A REPORT ON THE COMMERCIAL AND EDUCATIONAL APPLICATIONS OF EXPERT SYSTEMS

by

DAVID PETER DURANT

A thesis submitted to the
Faculty of Education
of the
University of Birmingham
for the degree of
MASTER OF EDUCATION

Department of Curriculum Studies
University of Birmingham
P O Box 363
Birmingham
B15 2TT
ENGLAND

March 1989

SYNOPSIS

Expert, or intelligent knowledge-based, systems have emerged as the main practical application of Artificial Intelligence research. This thesis reports on their history, development and increasing commercial application. An analysis of the tasks and domains of 785 systems is reported which indicated a level of task specificity. The technology is suggestive of significant educational relevance as it is closely linked with concepts of expertise, intelligence, knowledge and learning. These basic educational concepts are discussed. The thesis reports on a survey of the use of the NCC Expert System Starter Pack in Further and Higher Education. The relationship between other computer-based learning systems and expert systems are discussed and it is argued that the development of intelligent tutoring systems is a more complex operation than the educational application of expert systems. A wide spectrum of potential educational applications is indicated. It is suggested that placing pupils in the position of knowledge engineers provides an exciting curriculum application. It is further argued that the use of expert systems in a commercial training role promises to be a major future development. Other educational applications are considered and the wider social implications associated with the use of expert systems are summarised.

DEDICATION

nam et ipsa scientia potestas est

knowledge itself is power

Bacon (Religious meditations)

This work is dedicated to the memory of my father, Thomas William Durant, who died before its completion.

ACKNOWLEDGEMENTS

Some of the works noted in the references and bibliography have acted as information sources and others as catalysts to further thought. I acknowledge the debt I owe to all the authors. I also acknowledge my thanks to Peter Piddock, who initially supported my research ideas, and to Tony Fitzgerald, Ian Selwood and Dr William Wynne Willson who have read and commented upon this manuscript during its preparation.

I owe much to Dr Matthew Dixon, John Heslop, Linda Hughes and Mike Newman (staff at the National Computing Centre, Manchester). They gave freely of their time and knowledge and without their goodwill, the production of many aspects of this work would have been made much more difficult, if not impossible.

Finally I acknowledge the support that has been given to me by my family. To Sue, Neil, Steven and Christopher

'Dad has finally finished'

CONTENTS

INTRODUCTION

Expert systems, or intelligent knowledge-based systems, have emerged as the main practical application of artificial intelligence research. These systems promise, for different reasons, to be both major commercial and educational innovations. The commercial software industry has provided a range of tools for business use (word processing, databases, spreadsheets etc.) which education has applied and developed for its own purposes. Expert systems are suggestive of significant educational relevance as they are concerned with such fundamental educational issues as knowledge, intelligence and learning. The concept of computers and education in general, is primarily not an educational innovation, but rather an educational application of a technological innovation. Expert system technology provides another tool which is being increasingly applied in industry and, fortunately for education, this offers an exciting tool which may have important applications both inside and outside the classroom.

This thesis divides into three parts;

expert system technology and the associated educational issues,
the application of expert systems in the 'real' world,
and the educational applications of the technology.

In Part One the history of expert system development is traced and an assessment of what constitutes an expert system is provided. The use of the term 'expert' or 'intelligent' in the name has possibly raised expectations of these computer systems to unattainable levels. The limitations of expert systems are noted in chapter two.

An analysis of what is an expert and the nature of expertise is contained in chapter three. The issue of expertise and its acquisition, fundamental to any learning system, reappears in chapter thirty.

Intelligence and thought, both in machines and humans, is covered in chapter four. The computer has been compared to the brain and human qualities of intelligence and thought have been attributed to computers. Important distinctions are made between computer and brain and human and artificial intelligence. It is suggested in chapter five that the production of machines that 'think' like humans is unlikely as we still do not have a clear understanding of the thinking process in humans.

The constituent features of expert systems are identified in chapter six. The basis of any 'knowledge-based' system is knowledge and the differences between data, information and knowledge are examined. The methodologies involved in acquiring and representing knowledge, even if that knowledge is uncertain or incomplete, and creating an expert system using that knowledge are discussed.

Part One concludes with a comparison of conventional and knowledge engineering techniques and an assessment of how to go about building an expert system.

Part Two is concerned with the potential and actual commercial applications of expert systems. Chapter fourteen argues that it is easy to describe the tasks performed by various systems, but difficult to provide clear-cut distinctions between them. An assessment of Johnson's 'islands' hypothesis of task specificity is provided and it is argued that this analysis is of more relevance to industry than education.

An investigation of 785 systems is undertaken and the results are summarised in chapter sixteen. In addition to this analysis, chapter seventeen reports on the findings of other surveys into the applications of expert systems.

In the remaining chapters of Part Two,

details are provided of systems in a number of domains, but with particular reference to engineering, medical, financial and legal applications. In the latter two examples, the wider financial and legal issues of applying expert systems in any domain are also considered.

Part Three, concerned with the educational issues and applications, begins with a look at the educational use of computers in general. The implementation of educational applications of expert systems is at an early stage, but it is argued in chapter twenty five that placing the pupils in the role of knowledge engineers provides a simple, yet exciting potential. Details are provided of a number of projects.

The National Computing Centre produced, as part of the Alvey programme, an Expert Systems Starter Pack. A survey into the use of the Pack in Further and Higher Education is reported in chapter twenty six.

Although designed for commercial rather than training purposes, it was noted that during the use of commercial applications, there were training elements available to both system builders and system users. Chapter 27 suggests that the application of expert systems in this commercial training role promises to be a major future development.

In addition to expert systems, AI research may also offer a number of other educational applications including intelligent computer aided learning and intelligent tutoring systems. These are covered in chapter twenty eight where it is argued that the development of such systems is much more complex than that of developing expert systems.

As we progress to an increasingly information-based society where access to information is paramount, a vital developing role for expert systems may prove to be in intelligent retrieval systems. This is considered in chapter twenty nine.

Artificial intelligence research may, in the long term, provide a greater understanding of the teaching and learning process both by humans and machines. Although the educational applications of expert system technology is at a very early stage, their impact on learning is assessed in chapter thirty, as is research into machine learning.

Finally the relationship between man and machine, both on a practical and philosophical level, and the far reaching social implications associated with the increasing application of expert systems, both within and without education, are considered.

PART ONE

EXPERT SYSTEM

TECHNOLOGY

WHAT IS AN EXPERT SYSTEM ?

Expert system technology is a product of research into Artificial Intelligence (AI) and this opening chapter reports on the history of these developments. Despite the name, expert systems are not omnipotent systems and their characteristics and limitations are discussed, but beginning with a look at the definitions of expert systems.

Providing a set definition of an expert system is similar to aiming at a moving target, as various terms such as

rule-based system
knowledge-based system
intelligent knowledge-based system

are often used synonymously with the term 'Expert System'.

The use of the term 'intelligent' is open to debate, although it was included in the term chosen by the Alvey Report. Likewise the importance of the concept of the knowledge base is recognised in the Alvey name for this area of technology 'IKBS - Intelligent Knowledge Based Systems' (Alvey 1982).

In fact the actual word 'expert' in 'expert systems' is not an attributive modification, which means that an expert system is not simply a system that is expert at some task. Morrow, reported in Foremski (1986), wished that

> "expert systems were called something else such as competent systems to be more accurate and describe a more marginal role yet still a useful one. People have a tendency to be too ambitious with expert systems"

There are many computer systems that are 'expert', but not expert systems (for example an aircraft autopilot system) and many expert systems that are not very expert. This situation was appreciated by Buchanan and Shortliffe (1984) where the use of the term 'expert system' was seen as a pun, designating a system that acts as an expert on a major task and also as a consultant to someone who has that task. This duality could not last and as the population of so-called expert systems rapidly increased, the validity of the term became diluted to include all manner of products bearing the label of 'expert' or 'intelligent'. This excessive use of such terms, particularly, but not exclusively, by sales and marketing staff, was perhaps inevitable once 'expert' or 'intelligent' products began to gain a degree of commercial respectability. This may in turn have had a damaging effect upon the wider development of the technology, as it possibly had an off-putting effect on potential customers and developers of such products.

The use of the term 'knowledge' in the title does not presume any understanding of the program, even though it may appear to exhibit a degree of understanding in a limited domain. This can be achieved by subterfuge, cunning and clever programming. The term also implies some form of cognitive process at work, but the only cognitive processes involved will have been those of the knowledge engineers and the experts, at the stage of system building. At run-time the expert system will be on automatic pilot.

The use of such labels as 'knowledge', 'intelligence', 'expert' and so on imply that the computer assumes some form of human power over the knowledge. This is erroneous because computers do not do anything more than process data. As described in chapter seven, data is an inert commodity, whereas knowledge, residing in the human users, is active (the degree of activity depending upon the circumstances at the time). The confusion between 'data' and 'knowledge' leads to the false ascribing to computers of the power of consciousness.

What are the definitions of expert systems ?

There are almost as many definitions of an expert system as there are actual expert systems, but the definitions divide into two broad camps. There are those definitions, as noted by Johnson (1984), which take a theoretical approach

"high level emulation of the performance of a human expert"

and those which take a more practical approach

"applying the techniques of logical inference to a knowledge base".

The BCS Expert Systems Special Interest Group defined them, in July 1982, as follows

> "an expert system is regarded as
> the embodiment within a computer of
> a knowledge-based component, from
> an expert skill, in such a form

> that the system can offer intelligent advice or take an intelligent decision about a processing function. A desirable additional characteristic is the capability of the system, on demand, to justify its own line of reasoning in a manner directly intelligible to the enquirer"

Notice that the definition does not mention decision making, but merely the giving of advice to assist in the decision making process. Alty (1985) defined them as follows

> "..a computer program which
>
> 1) aims to emulate (or perhaps even out-perform) the thought processes of one or more human experts in a skilled diagnostic or other decision making task.
> 2) explains its conclusions or decisions to the user on demand."

Likewise, Sell (1984) and Bramer (1984) both stress the 'human expert' aspect

> "a knowledge-based system that emulates expert thought to solve significant problems in a particular domain of expertise"
> (Sell 1984)

> "a computer system which embodies organised knowledge concerning some specific area of human expertise sufficient to perform as a skilful and cost effective consultant"
> (Bramer 1984)

Note, however, that Bramer does not explain how to determine whether a system is skilful or not. Addis (1982) views expert systems as degrees of enhancement to an information retrieval system, since the competence of a human expert is often dependent upon complex retrieval skills.

Although Alty (1985) included the word

'program' in his definition, the use of the term 'expert system' is justified rather than calling the system just a program, as it ~~does~~ contain both a problem solving component and a support component. Later in this thesis (chapter twelve) I will be discussing the difference between traditional application programming and expert systems.

The Alvey Report (1982) uses a much more simple definition

> "an expert system is a system that
> uses inference to apply knowledge
> to perform a task"

D'Agapeyeff (1983) stressed their problem solving nature in that they solve substantial problems generally conceded as being difficult and requiring expertise. They are called knowledge-based because their performance depends critically on the use of facts and heuristics used by experts. Feigenbaum (1982), one of the founding fathers of this technology, applied the following summary definition

> "an expert system is an intelligent computer program that uses knowledge and inference procedures to solve problems that are difficult enough to require significant human expertise for their solution. The knowledge of an expert system consists of facts and heuristics, the facts constitute a body of knowledge that is widely shared, publically available and generally agreed upon by experts in the field. The heuristics are mostly private, little discussed rules of good judgement that characterise expert level decision making in the field."

Although so many attempts at a definition have been made in the literature, Chandrasekaran (1983) has shown that each of the definitive characteristics of expert systems is displayed by systems which are not generally acknowledged to be expert systems or that each is missing from many systems that are acknowledged to be expert systems.

What are the characteristics of expert systems ?

The lack of a standard definition of an expert system could lead to the situation where they mean all things to all men. Indeed, this may have some truth in it. Essentially an expert system consists of a knowledge base and an inference mechanism. The inference mechanism looks in the knowledge base to see which rules are satisfied, selects one and fires it to perform the corresponding action.

Although there are a variety of definitions of expert systems, there appears to be a consensus agreement upon the qualities that expert systems should possess. The declarative rather than procedural style of programming remains a distinctive feature.

Rychener (1985), Buchanan (1986) and Waterman (1986) all identified performance and expertise, defined as exhibiting high performance with a high level of skill, even though this may only be possible in a narrow domain, as being one important characteristic.

Goodall (1985) observed that expert systems do not attempt to use a mathematical representation of the problem, even if such an approach was possible, but apply procedures to reason with symbolic information and use heuristic inference. As noted by Buchanan (1986), further characteristics are observable in their implementation, being quickly alterable with a comparatively low risk of unwanted side effects and the ability to grow gradually by adding new pieces of knowledge, usually in the context of solving an unfamiliar problem.

Alty and Coombs (1984) have argued that expert systems are just a development of traditional data processing. However, expert systems are different and they also differ from other AI applications in that they perform tasks at expert levels of performance and Newell (1968) recognised that they emphasise domain-specific problem strategies over the more general weak methods of AI.

Although many of the early systems have been designed to be used by 'experts', it is likely that in the future they may be designed to be used by 'non-experts', it is an essential feature that expert systems are able to explain their decisions. D'Agapeyeff (1984) identified the fact that they possess 'understandability', in that they are readable by those who provided the knowledge and potentially by similarly knowledgeable users and managers. This ability to provide explanations of their reasoning on demand and the ability to explain and justify answers either on the basis of theory or by citing relevant heuristic rules is an important characteristic. The systems employ self knowledge to reason about their own inference processes and provide justification or explanations about conclusions that are reached. It is this latter characteristic that has the exciting potential for further development and future applications. Present systems are limited in this direction as they simply display the rules that led them to the particular conclusion, but future systems could become highly adept at analysing their reasoning processes and construct rational lines of argument tailored to fit the specific user.

Explanation strategies usually display the inference strategy of the system, rather than of the experts, who frequently use analogy when explaining their reasoning. This may be adequate in explaining how the system arrived at the conclusion, but is inadequate as a teaching methodology.

The early ideas

The concept of expert systems is not a new one. McCarthy (1958) proposed the creation of an advice taking system that could accept advice and make use of it to plan and execute actions. Until the late 1970's there were few attempts to write programs that could learn by taking advice. The recent emphasis on expert systems has focussed new attention on the problem of converting expert advice into expert performance.

In the early days computers were seen as just big, fast calculating machines. As storage capacity increased it became apparent that they could do more than just store, modify and retrieve data, it was realised that the machine could recognise patterns. Human experts recognise patterns by seeking similarities or differences from previously recorded patterns. If the knowledge of the expert could be committed to a computer, then the machine could act as a quasi-expert. Furthermore, if all the experts in a particular field could commit their skills to the machine and constantly update the knowledge, then if the system was made accessible to others, then the general level of expertise would rise and expert information could be provided on a much wider scale than at present. This, it should be stressed, is a very ambitious aim.

This area of technology investigates methods and techniques for constructing man-machine systems with specialised problem solving expertise. Researchers have tended to put the emphasis on knowledge rather than on formal reasoning methods, because many problems do not have an algorithmic solution due to their complex context which generally defies precise description and rigorous analysis.

Most 'intelligent' programs are single-minded experts within their single domain. MYCIN (Shortliffe 1976) is not intelligent in the classical sense because the reasoning has been laid down by the programmers as a set of facts each with a statistical weighting. This is similar to the action of a doctor weighing up the evidence for the likelihood of the diagnosis being disease A or disease B. The advantage of the computer is that it can store vast amounts of facts in memory and work at a fast rate.

<u>What are the limitations of expert systems ?</u>

An expert system is no different from any other computer system in that every computer system has its theoretical and practical limitations and often these limitations provide opportunities for further research. A system builder in a particular domain may come across a particular limitation which can only be solved by studying the theoretical issues being addressed by another research worker. Over the last decade, some of the early limitations have diminished, and we have moved to a point where rule-based systems are relatively easy to build. This movement has only been possible as a result of the earlier work of many AI scientists.

Nevertheless, expert systems do have specific limitations, in particular, as noted by Buchanan (1982), they must operate in a restricted domain and they are unable to recognise the limits of their ability. Additionally, although some systems may be able to handle uncertain or incomplete data, they cannot easily deal with inconsistent knowledge.

This restriction is not confined to the size of the application domain, the systems have a restricted, and often stylised, language for input, output and explanation, a feature which is discussed later. There are further limitations regarding the representation of rules and information which are also discussed in more detail in chapter nine.

An important limitation, noted by Hart (1980), is that there may not be any independent means of checking whether the conclusions obtained from the system were reasonable. (Caveat emptor !)

Basden (1983) suggests that the limitations will always be governed by actual specific applications and also, I would add, by user expectations. Even though they have limitations, expert systems can be applied successfully in many domains. It is the responsibility of the user to select a suitable system bearing in mind, among other things, the problem and the system's particular strengths and weaknesses. However, a layman may not be aware that an expert system has limitations and lacks one or two rules. It may be the case that the expert is also unaware of this deficiency. In a commercial environment, this may cost far more than employing a human expert in the first place.

As it is necessary, for technical reasons, to restrict the amount of information contained within a system, it will therefore have limitations. This limit will be established by what has been made explicit within the system. It is essential that users are made aware of such limitations. This is a particular problem associated with the use of the term 'expert' system whereby users may overestimate the capability of the system.

Warnier (1986) argued that

> "the computer is not a substitute for the human being, but rather a tool to be used by human beings"

As suggested by Speller and Brandon (1986)

> "there ought to be a point where expert systems are accepted as assistants and not as oracles."

Whatever the limitations of expert systems, real and imagined, the advent of 'intelligent systems' has been forecast by Hayes-Roth, Waterman and Lenat (1983).

> "machines that lack knowledge seem doomed to perform intellectually trivial tasks. Those that embody knowledge and apply it skillfully seem capable of equalling or surpassing the best performance of human experts"

Having considered what constitutes a computer-based expert system, it is now necessary to undertake the same exercise for human experts.

WHAT IS AN EXPERT ?

As has been shown in the previous chapter, it is difficult to provide an accurate definition for an expert system. By the same token, it is difficult to provide one for an 'expert'. Everyone will have their own perception of an 'expert' and could recognise the work of one, but would find it difficult to put it concisely into words. Part of the problem is that terms like 'intelligence', 'knowledge' and 'expert' are themselves ill-defined. This analysis leads to several related questions.

What makes an expert, expert ?

Hartley (1981) defined experts as

> "those who define what the work is and how it is to be carried out".

Hartley further notes that 'experts' are always a minority group, and are only 'expert' relative to the 'practitioners' who are those who use the system, created by the 'experts', in the course of their everyday work. The answer to the question of whether a single expert is necessarily better than a large group of practitioners, must depend upon the circumstances and the nature of the expertise. If the knowledge is structural or systemic in nature then the answer must be 'yes', because experts are expert because they have the ability to perceive the structure of the domain and not just its content. However, if the knowledge is fragmentary and largely consists of facts which are additive in nature, then the answer is likely to be 'no', although the precise answer will depend upon the number and quality of the practitioners. In this case, the expert only has his own experience to call upon and while this may be considerable, it is unlikely to be better than the aggregated experience of a large group of practitioners.

The 'expert' members of society do not appear to do anything different from the rest of society, other than they have the capability of comprehending more relevant information than the layman and are more aware of inherent processes and possible implications. Experts were characterised by Hawkins (1983) for their efficiency, effectiveness and an awareness of their limitations (the 'half of being smart is knowing what you are dumb at' syndrome).

Hayes-Roth, Waterman and Lenat (1983) identify four features of an expert;

quality of performance, it is of no benefit in making the wrong decision,

speed of decision making, it is no good taking all day to make a decision, although it may not be 'pure speed' but rather the ability to make a decision as against not being able to do so,

explanation, the ability to provide full details of how the decision was reached, not just a trace of the expert rules,

there will also be a 'trade off' between depth and breadth of specialisation, as one can only know a lot about a little. Chapter nine contains a discussion of this deep and shallow concept.

Johnson (1983) defined experts as

> "persons who, because of training and experience are able to do things that the rest of us cannot; They are not only proficient but also smooth and efficient in the actions that they take.
> They know a great many things and have tricks and caveats for applying what they know to problems and tasks.
> They are good at ploughing through irrelevant information in order to get at the basic issues
> They are good at recognising new problems they face as instances of types of problems with which they are familiar."

The Concise Oxford dictionary definition is

> "someone who has acquired a special skill or knowledge in a particular subject"

Hawkins (1983) sees as 'expert' someone

> "who can negotiate an agreed interpretation of a particular subject with the help of special knowledge and user opinions".

This means that an expert could be used as an analytical tool helping users make well-informed decisions without forcing them to accept any particular interpretation or procedure. However, as will be discussed in chapter thirty two, experts also have the power to intimidate.

<u>What is the relationship between experience and expertise ?</u>

There is experience in two forms;

experience OF (having observed a situation)

experience IN (having done something, the ability to solve problems).

It is this latter form that has been used in expert systems, often in the form of IF-THEN rules. Experts are experts because of their acquired knowledge and what they are able to do with it. Goodall (1985) observed that not only do they have this body of knowledge which is unfamiliar to the layman, but a proven record of being able to use that knowledge.

Experts have 'private' knowledge, as a result of experience, which is in addition to 'public' knowledge as contained in textbooks. It is this knowledge which is not only crucial to their daily work, but to the success of building a system based upon their knowledge.

Waterman (1983) identified the differences between human and artificial expertise

human expertise	artificial expertise
perishable	permanent
difficult to transfer	easy to transfer
difficult to document	easy to document
unpredictable	consistent
expensive	affordable
creative	uninspired
adaptive	needs to be told
sensory experience	symbolic input
broad focus	narrow focus
common sense knowledge	technical knowledge

Human expertise is probably not yet understood to a sufficient degree for users to specify what may be needed of, or expected from, an expert system.

<u>What kind of understanding capability does an expert have that a novice doesn't possess ?</u>

As noted by Chi et al (1981), Kolodner (1983) and Barfield (1986), an expert is more knowledgeable about his domain and knows how to apply his knowledge more effectively than does a novice. Perhaps a good example involves the oft-quoted story of a man who repaired a car by simply giving it a kick. The owner complained bitterly when presented with a bill for £100. The repairer then presented the owner with an itemised bill;

to kicking machine	£ 1
to knowing where to kick	£ 99
	£100

In this case the definition of an expert was a man who knew where to kick. I can sympathise with the above car owner, as I have this knack of causing more problems than previously existed whenever I venture underneath a car bonnet. Even with my toolkit and a selection of manuals and reference books, I have realised my own limitations in this direction and now accept that I need expert help (and incidentally the need to pay for this assistance). However, this realisation of my limitations does not make me an expert car mechanic, although it may link to the 'half of being smart' syndrome, as mentioned earlier. We use experts in a mix of information providers, problem solvers and explainers. As regards my car, the function of the expert may be

a) explaining; what to do

b) informing; where to go to find help

c) problem solving; as is often the

case, just doing the job. This may be because it is easier for the expert to actually do the job rather than explain how to do the job.

The expert will be paid for his services, after all that is how he earns his living. There is a link here, which is discussed in chapter eight, to the resistance of some experts to give up their knowledge.

Duke (1985) considered the factors that should be considered important when seeking expert help. He suggested;

a) familiarity with an appropriate and extensive database,

b) shrewd analytical abilities and diagnostic skills,

c) predictive abilities based on sound judgement (whether objective or subjective),

d) presentational and explanatory abilities (this is vital to demonstrate the level of confidence that may be placed on the findings),

e) successful record and high reputation findings),

e) successful record and high reputation (this is dependent upon the reliability of recent judgements).

In short, the expert should be furnished with a comprehensive, up to date, knowledge base and should be continually acquiring new information as it becomes available.

<u>What kind of understanding does an expert have that an expert system doesn't possess ?</u>

Michaelsen et al (1985) and Davis (1982) provide similar lists of capabilities, stating that experts are capable of;

a) applying their expertise to the solution of problems in an efficient manner

b) employing plausible inference and reasoning from incomplete or uncertain data

c) communicating well with other experts and acquiring new knowledge

d) restructuring and reorganising knowledge

e) breaking rules (experts have almost

as many exceptions as rules and they understand both the spirit and the letter of a rule)

f) determining relevance, knowing what information is important in a given situation

g) degrading gracefully and knowing when a problem is outside their domain. At the boundaries of their domain, they gradually become less proficient at problem solving rather than coming to an abrupt halt (compare the slope of a hill and a brick wall)

Expert systems have only achieved the first three capabilities and there is some argument about the quality of man-machine communication.

Feigenbaum (1979) noted that human problem-solving behaviour is weak and shallow, except where the human expert is a specialist. However, the transfer of expertise between speciality areas is slight. The expert chemist would be a specialist in a small sub-set of chemistry, but is not necessarily, also a chess master and an engineering expert. This observation is the basis of the development of many modern expert systems which work in a highly restricted application domain. The performance of these systems is based on knowledge about a particular domain, rather than expertise in general problem-solving.

Feigenbaum and McCorduck (1984) observe that

> "a human expert solves problems all right but he also explains the results, he learns, he restructures his own knowledge ....
> part of learning to be an expert is to understand not merely the letter of the rule, but its spirit ....
> he knows when to break the rules, he understands what is relevant to his task and what isn't. Expert systems do not yet understand these things."

The nature of expertise and its acquisition is a basic consideration of any learning system and this issue is continued in chapter thirty. Attention now turns to other basic educational concepts, starting in the next chapter with intelligence and thinking.

<u>INTELLIGENCE AND THINKING</u>

Should the starting point for a discussion on the nature of intelligence be from the viewpoint that intelligence is a purely human trait, or from the computer science viewpoint that intelligence characterises all information processing systems ? There is a degree of overlap between the proposition that intelligence is a mysterious and somewhat elusive facility which relates to mental ability in the human cognitive process and the proposition that intelligence comprises a set of abilities or attributes functioning as a complex system. The overlap was demonstrated by Newell (1980)

> "all intelligent activity is based on search"

This was followed by Berliner (1981), working in the domain of chess, suggesting that

> "knowledge without search has a limited utility as has search without knowledge. For each domain

> a certain balance appears to
> exist."

Fischler and Firschein (1987) commented that AI research has indicated that intelligent behaviour requires stored knowledge and means of manipulating that knowledge, but that the relationship between the encoding or representation of the knowledge and the purpose for which that knowledge is to be used is critical.

Dictionary definitions of 'intelligence' frequently include features such as 'knowing, reasoning and understanding' Intelligence may be easier to recognise than define, the word acquiring a number of meanings and implications. Guilford et al (1956) noted that

> "much easier to decide what general
> reasoning is not, than to say what
> it is ....
> general reasoning has something to
> do with comprehending or

structuring problems of certain kinds in preparation for solving them ....
it may be a general ability to formulate complex conceptions of many kinds."

Intellect is a factor of the number of differences that can be handled or dealt with. The test of a first rate intelligence has been defined by Chalmers et al (1971) as

"the ability to hold two opposed ideas in mind at the same time and still retain the ability to function."

Feigenbaum's definition, reported in Boden (1987), is that

"intelligent action is an act or decision that is goal-oriented, arrived at by an understandable chain of symbolic analysis and reasoning steps, and is one in

which knowledge of the world
informs and guides the reasoning"

Sloman (1987) took a more pragmatic approach defining human intelligence as

"productive laziness"

meaning the sharing of tasks between brain and computer. For example, saving time by making more of the results of previous experiences available (computer memory and human memory). He envisages the development of intelligent front ends enabling humans to solve more difficult problems much more quickly.

Although there are diverse conceptions of intelligence, Sternberg (1986) describes the common features contained in two dozen definitions including;

a) noticing similarities between events

b) making generalisations

c) the ability to learn

Intelligence is not purely a property of homo sapiens, it is acceptable to accord 'intelligence' to many species, but there is not a clear cut-off point on the biological spectrum below which behaviour could be considered as unintelligent. Techniques to make meaningful comparisons between human intelligence and that of other biological or mechanical systems have yet to be developed. Similarly it cannot be assumed that human intelligence is at a point above which intellectual skills cannot rise.

As time passes, computer systems are becoming more 'intelligent' and the reality of systems which pass the Turing Test (Appendix 2) is no longer a mere dream.

There is a fundamental question in that 'how far is it possible to create an 'artificial' intelligence when our understanding of 'real' intelligence is so limited ?' AI researchers are divided into two camps, the 'scruffies' who believe that it is premature to attempt to formalise human reasoning tasks because human intelligence is a multi-faceted ability and our present knowledge of these facets is very limited. On the other hand, the 'neats' preserve an open mind believing that there are general principles of intelligence to be discovered, even if human intelligence does not actually operate on those principles. As a consequence, AI research has tended either to produce machines that perform tasks which humans need to do or to analyse intelligent human behaviour in terms of information processing. Some AI research focusses on 'intelligence', while other work is concerned with 'artificial'.

It is perhaps a measure of the developments in AI research that there should be a shift in the definitions of AI provided by Boden (1977, 1987)

> "the study of how to make machines do things that would require intelligence if done by people" (1977)

> "the study of how to build and program machines that can do the sorts of things which human minds can do" (1988)

An indication of the complexity of research in this area is provided by Butcher (1973) who noted that

> "the study of human intelligence has yielded a large accumulation of knowledge about individual differences, but very little about the basic laws of cognitive functioning"

Estes (1982) further noted that

> "it would seem that progress towards untangling the multiple determinants of individual differences in intelligent behaviour can come only within the framework of more comprehensive theories of the whole interactive cognitive system"

Is intelligence content or process ?

Rutowska (1986) argued that human intelligence cannot be understood solely in terms of internal structures and processes without consideration of the many social and physical environmental factors. Additionally, any intelligent system must be able to modify its actions as a result of information received from sensors.

A description of intelligent activities by Dennett (1979) seemed to confuse 'process' with 'product'. No 'product' can be intelligent, it is the 'process' which requires intelligent decisions. However, try telling that to the sales directors of the companies producing the variety of 'intelligent' products on sale today, including a 'higher intelligence' putty (Star Tack made by Seal Strip Ltd UK).

There is more to human intelligence than a number of production rules. Fischler and Firschein (1987) describe both the attributes of an intelligent agent (eg learning, planning, understanding) and those related to, but distinct from, intelligence (eg emotion, aesthetic appreciation, muscular coordination). Human beings frequently behave unreasonably, the world of politics provides ample examples of such behaviour and human behaviour, in general, is affected to varying degrees by cultural and social factors. This 'unreasonableness' may be linked to ideas of creativity and genius. Human learning is not purely a reasoning process, but proceeds through intuitive leaps, common sense and lateral guess work.

Torrance (1986) comments upon the nature of intelligence and while accepting the plausibility of simulating human cognitive activities on a computer, he is concerned about the wider AI claims of any mental state being able to be simulated by computer and therefore being computationally explicit.

It may be that human intellect is a 'spaghetti-like' collection of heuristics and that the collection of each individual is unique. If this is the case, then the attempts to build and implement human intelligence on machines is probably further beyond us than is presently accepted. One reason put forward for AI research is to further understand our own intelligence. If there are no basic common principles between different intelligences, then any 'successes' at developing machine intelligence, may not provide us with the insight which we seek, but rather provide a copy, or copies, of human intelligence.

Negrotti and Bertasio (1986) distinguished three levels where the human processing of knowledge takes place;
a) the brain - processing signals
b) the mind - intelligent processing of information according to established rules
c) the intellect - knowledge activity, 'reason'
They note that AI research can only claim

success at level (a) and limited success at level (b).

Schank, quoted by Durham (1986), argued that

> "if you are looking at a psychological system, why assume that the mathematical systems that were invented for other purposes happen to fit neatly ?"

There are many examples of successful computer programs that have been written to solve specific problems. There have also been attempts to write programs, with less success, that can solve general problems, for example GPS (Newell and Simon 1963). Human problem solving involves general knowledge (common sense ?) over and above the specific knowledge required for the solution of any particular problem. Success at some task is a function of specific skills, experience and 'intelligence', but it is specific to the said task.

The specific nature of machine intelligence, compared to that of humans, can be shown by comparing performance at Chess, a task widely used in AI research. Chess programs have been developed which operate at 'Master' level and such programs can defeat all average to good human players. However, those chess programs have no general intelligence outside their limited domain. Human skill at Chess is a function of general intelligence, experience and specific skills and because of this general component, an average to good chess player is likely to also be able to play a good game of, say, Bridge.

An artificial device must be able to communicate freely and effectively if it is to be considered 'intelligent'. There is not a sharp division between intelligent and not intelligent, there will be a graded series of devices with varying levels of 'intelligence'. The mice that are developed for use in maze-exploration contests, can perhaps be considered as intelligent within a narrow domain and their designed limitations.

As a maze is essentially a tree structure, it is not a difficult task to write suitable software. These mice can operate in 'exploration mode' where they patiently map out the various passages and dead ends and 'learn' about the maze. They can then use 'race mode' where they use the information gained during the exploration phase to enable them to traverse the maze in as short a time as possible. However, a maze is a very closed situation with limited alternatives and a finite amount of data is required to find a solution. Genuine intelligence, not just artificial intelligence, entails dealing with any unexpected situation. If between the two phases, the maze is altered, then the mouse with more adaptability (intelligence ?) would be more likely to get through the maze than the 'low intelligence' mouse who has become stuck and perhaps reverted to 'exploration mode'. Even the decision to revert to 'exploration mode' could be considered as an 'intelligent' action.

No set of computer program instructions can completely capture the infinite complexity of the world and, as discussed by Lehman (1988), even if a program is successful today, it may be invalidated tomorrow because the world is a continually developing and changing place. It is a measure of man's intelligence that the species has survived for so long within that changing world.

Thinking

Cogito, ergo sum (I think, therefore I am)
Descartes (1596 - 1650)

Thinking is a complex process, Schank (1982) notes that possibly the most significant advance made in the last decade is the appreciation of the level of this complexity. Boden (1983) proposed that a theoretical aim of AI should be to specify the procedural complexity of thinking. However, as noted by Schank and Hunter (1985)

> "the quest to understand thinking begins not with complex issues but with the most trivial of processes"

This statement is relevant because much of the early AI work in this area concentrated upon tasks, such as chess, which were thought to provide suitable examples. The techniques developed by these workers was discovered to be not the same ones that were used by humans. Hence they began to attack the problem at a different level by attempting to produce computer programs which could handle tasks that humans would consider trivial.

de Bono (de Bono 1982) defined the skill of thinking as

> "the operating system with which intelligence acts upon experience for a purpose."

However thinking is not intelligence in action. The nature of intelligence in action can be perceived as an efficient system of information receiving, an ample and efficient information store that can modify its storage, its storage system or its processing methods as a result of its experiences and some means of communicating its decisions to the outside world.

The first three of the above probably correspond to Piaget's theories of assimilation and accommodation. Highly intelligent people may not be good thinkers, nor are they automatically good thinkers. Indeed Turing (1950) put forward his test for a 'thinking' machine and not for an 'intelligent' machine, realising that a thinking machine would have to be intelligent, but that an intelligent one might not be able to think.

Feigenbaum and McCorduck (1984) observed that

> "almost all the thinking that professionals do is done by reasoning, not calculating"

We are ignorant about human thinking and AI only suggests rather than defines the information processing details of human thought. Boden (1983) suggests that AI can help us to understand and improve thinking and reported that researchers have spent much time considering highly intelligent behaviour such as playing chess, solving complicated problems and proving mathematical theorems, but discovered that the techniques that they developed were not the same ones that people used to perform the same tasks.

The history of search for the 'rules of thought' which began as far back as Plato, was traced by Dreyfus (1979). Johnson-Laird and Wason (1977) report of psychological attempts to abstract rules in the form of 'effective procedures'. The attempt to explain human cognition in terms of rules was refuted by Dreyfus and Dreyfus (1987) because, except in the case of a complete novice, skill is not just a case of the mechanical application of rules, skill is only developed as a result of practice and experience. All parents will recall the amount of thought and concentration that their children have to put into learning the skill of, say, being able to tie shoelaces.

Once the skill has been acquired, it becomes 'automatic' and needs little further conscious thought.

Norman (1981) argues that analyses of human performance imply a class of processing structures that is quite different from that which is commonly envisioned by AI. The virtue of the computer is speed and power, whereas the virtue of a human is creativity and flexibility. Human reasoning possesses capabilities such as the ability to pursue new and unforeseeable lines of reasoning in response to a new situation and the ability to recognise what information is not present. Expert systems have the ability to 'ask' for missing information, but only to fill undefined variables in the knowledge base which is not the same as seeking 'missing' but 'unknown' information.

Humans perform many different things at a time and use different processing structures. For example, while driving a car along a busy road, in itself a multi-tasking operation, the driver is often involved in other tasks that are not directly connected with the task of driving the car, including conversation with the passengers and interaction with the external surroundings (scenary, shop fronts and pedestrians along the pavement). In computer terms this concept of multi-tasking or multi-processing means that either the system will have sufficient processing power so that the tasks can be handled separately without any interaction or interference or the system will need to constantly switch between the tasks, alternately saving the status of one task, switching over and processing another task before saving that and switching over again.

In human terms, when we lack the necessary processing power, we delay and defer goals and actions as appropriate. To pursue my driving analogy, many road accidents may be the result of such inappropriate delay or postponement or failure to postpone some other task.

The question of how should a machine think presupposes an answer to the question of whether a machine can think. Some philosophers and psychologists would claim that the concept of 'thinking' is not a useful one, in fact preferring not to use the word at all. This is epitomised by Skinner (1971) who feels that discussion of inherently unobservable mental operations cannot possibly be the basis of either a science or a technology. The question of whether a machine can think is bound to provide an arbitrary answer.

A somewhat different question is that of 'does a machine think like a human ?' The temptation here is to believe that if a computer program mimics human behaviour then that program is a model of human behaviour. This argument (Simon 1972) has motivated much of the modern research on computer simulation.

No adequate quantitative theory of human feeling has yet been produced. It may well be that it never will, because feeling appears to be of a wholly different nature to thinking. We are often at a loss to explain our actions when based upon emotional feeling, whereas we can usually explain our actions that are as a result of thinking.

THE COMPUTER AND THE BRAIN

After having discussed the concepts of thought and intelligence, consideration is given in this chapter to the site of all this activity.

There has been a long standing discussion, as to whether the human brain functions in the same way as a computer. Shannon (1937) used Boolean algebra to describe the behaviour of relay and switching circuits. His argument was that if the laws of thought could express the behaviour of electronic circuits, then electronic circuits could express the laws of thought. The assumption here is that there is a correspondence between the behaviour of the neuron and the 'on-off' behaviour of the electronic switch. A prediction of the performance of a computer can be achieved by a careful examination of the computer circuits, but it is not possible to examine a brain in this way. Nevertheless, Looney and Alfize (1986) are aiming to produce an expert system on a chip. Their argument, going back to the original work of Shannon (1937), is that if the brain functions by firing or not firing particular synapses in Boolean fashion, then it would simplify knowledge representation to efficient ANDing and ORing.

Searle (1984) argued that human minds possess a quality of 'intentionality' that no computer can reproduce, but it has not been explained what in the structure of the brain could account for such a difference.

The initial concept of cybernetics involved a feedback system, the action of the system depending upon interaction with the system's environment. Aleksander, reported by Colley (1984) maintains that

> "cybernetics is not an attempt to make a human, its an attempt to pick up ideas from the human mechanism for engineered mechanisms."

The human brain and the electronic brain were viewed by Maugh (1986) as two different types of computers. However, consider the following two tasks;

a) evaluate the square root of 926754

b) ur brane, 4 xampl, wil probly hav litul trubl undrstndng ths sntns.

The computer would excel at the first task, but would certainly struggle to comprehend the second sentence, whereas the performance of the human brain would be in complete contrast. Many data processing programs work by employing strict pattern matching techniques which become less successful as the 'pattern match' becomes 'fuzzy'. An exercise that I have successfully undertaken, although it must be stated that it was a tedious business, is manually to match names of pupils that have been entered on computer marked test papers. The student, instead of writing the letters, 'writes' his name by colouring the appropriate lozenges. In each of the following pairs of examples, the names refer to the same pupil.

SMITHJOHN	VILLIAMSIAN	JONESSTEVEN
JOHNSMITH	WILLIAMSIAN	JONETTUEWEM

I was able to perform this task because of my experience and by using common sense, but a computer was unable to perform the task.

Likewise, in an example restricted to the names of football team stadiums. If asked to name where Tottenham Hotspur play, this is a piece of information that a human or expert system may be able to recall from memory or find in a suitable reference book (or knowledge base). If then asked to name where Hull Kingston Rovers play, an expert system would search through its knowledge base for the information and upon not finding it (as Hull Kingston Rovers are a Rugby League Club, not a Football Club) would suspect that the knowledge base was incomplete and ask the user for additional information. The human user, knowledgeable about sport, should not have much trouble in being able to spot the discrepancy.

It is interesting to note the following set of figures (with an accuracy of + or - 30%) produced by Chase and Simon (1974), relating to the human brain.

rate of information transmission along any input or output channel = 30 bits/second
maximum amount of information explicitly storable by the age of 50 = 10^10 bits
number of mental discriminations during intellectual work = 18/second
number of addresses which can be held in short term memory = 7
time to access an addressable chunk in long term memory = 2 seconds
rate of transfer from long term to short term memory of successive elements of one chunk = 3 elements/second
Note 1 chunk = 7 bits (Miller 1956).

The average brain has four billion neurons which make connections with other neurons via synapses. The brain's neuron network processes information by building patterns of communication between neurons. The neurons of the brain work in the order of milliseconds which is sluggish when compared with the computer components which work in the order of nanoseconds. Paradoxically, the brain does some things much faster.

Conventional Von Neumann computers which carry out predefined serial instructions do not operate in the same fashion as the human brain, but some computers are being designed to operate like the brain. However so little is known about the functioning of the brain that any attempts to develop vast neural network systems will probably raise more questions than answers. Even if a vast system was developed, it would only represent a tiny fraction of the connections that are available in the brain. The limitations of the machine as a brain were recognised by Andree as long ago as 1958

> "A computer is not a giant brain .... it is a remarkably fast and phenomenally accurate moron."

The brain must be rich in a variety of

structures and these structures must have a degree of complexity before it can become self modifying in any way that could be considered 'intelligent'. There are two main types of neuron in the human brain. Excitatory neurons, which are in the minority, push information forward. The function of the more common inhibitory neurons is to act as a screen or filter. Hence the structure of the brain is such that only significant information is allowed to pass into the processing areas. At birth we possess sufficient inhibitory neurons, but unless they are exposed to the information they will not develop. Neurons cannot regenerate and any that are not formed, or have been atrophied, cannot be recreated. This highlights the importance of a rich environment. Blakemore (1977) has shown that memory is spread around many parts of the brain. The more parts of the brain that are involved in learning a particular task, the better the memory. This may explain why the most forgetful people still remember how to walk or play music. The reason that memory can survive quite severe brain damage may be related to the way that the brain stores information.

The activity of the brain is a series of parallel processes and such activity could not be simulated on a single processor serial machine. However with the advent of parallel processing, neural network and transputer-based machines this hardware restriction would appear to be no longer a restriction. In parallel computers, processors may be connected to a number of other processors. However, the brain, with its four billion neurons, has a much higher degree of parallelism than this. Indeed the problem with parallel processors appears to be related to programming them rather than building them. It must be further noted that the brain has not been programmed by anyone, but programs, or fine-tunes, itself as a result of experience.

For a number of years researchers have been studying 'intelligent' computer systems in an attempt to gain further understanding of human intelligence. Looking at this from another perspective, it remains to be seen whether human intelligence can continue to comprehend the workings of computer simulations as they grow in size and complexity.

However there are limits to the amount of data that a human can comprehend simultaneously and thus a limit to the machine memory that can be used if the workings are to be intelligible to humans. Equally there is a limit to the computing speed of a human and thus a limit to the computing power that can be used if the program is to remain workable by humans. If these two limits overlap, referred to by Michie (1982) as the 'human window', there is a possible range of memory/computing combinations within which acceptable solutions lie.

The hippocampus is the part of the brain that is concerned with short-term memory and acts as a temporary storehouse between experience and long-term memory. In practice the brain filters out, or forgets, far more than it remembers. This is essential; if it didn't happen then the neocortex would quickly become swamped with information. It has not been explained how the hippocampus decides upon the level of significance of a new piece of information.

When new information is presented to the brain, it is capable of associating it to a number of related memory networks and constructing new linking systems. This is an area where computer technology may develop, but at present computers cannot handle information unless it is provided in a suitably precise form. The human ability to understand a situation comes from our ability to compare it with previous relevant situations. Tulving (1972) first proposed the distinction between memory which we gain from experience (episodic memory) and semantic memory that we use to understand the situation. Schank (1975) argued that such a distinction must be false, since both involve the same knowledge which must have been obtained by experience. Nevertheless this distinction has influenced such notions of cognitive structure as Merrill (1983). The SOPHIE systems (Brown et al 1982) capitalise upon episodic memory, using the experiences gained by the student, through problem solving activities, as the basis for directing further learning.

The machine works by building up a data structure and then compares this with the presented example. Any differences found are then incorporated into the data structure. Near misses must be catered for (fuzzy matching) otherwise the program would be overwhelmed by all the mismatches and be unable to work out how to modify its understanding. A schema is a data structure for representing a situation. The program understands the situation by retrieving an appropriate schema from memory and adapting it as necessary. Related schema are linked together to form 'schema-systems' eg sharing parts of their structure or by specifying transformations from one schema to another if used to represent actions or cause and effect. Learning is then interpreted in terms of the storing and modifying of schemata as a result of experience. This is Piaget's idea of structure (Piaget 1971); developing intellect by organising schemata and building on them to develop higher level structures and thinking involves the processing and changing of symbolic structures in memory. Piaget stressed the spontaneous interaction with the environment by which mental growth occurs and concluded that the main task of the teacher is to foster conditions under which each child can think freely.

It may be the case that the expert system concept of separating the knowledge from the inference mechanism corresponds to the basic organisation of the human mind. The brain works by relating events, while listening to a speaker, the brain may be subconsciously completing sentences (correctly or otherwise), agreeing, disagreeing or being ambivalent to the speaker's ideas or even considering some unrelated issue such as deciding what to have for tea. Some of these cognitive processes will have been the result of the random stimulation of ideas and others as a result of some link between what the speaker may have said, or done, and some event stored away in memory.

Bringing the appropriate piece of knowledge, out of a vast and constantly changing store, at the right moment in time, is the task of memory. Therefore memory is a vital component of cognitive activity. Schank (1982) notes that reminding is a powerful technique with which to investigate the structure of human memory. Often a small stimulus can trigger the recall of a whole series of memories. Memory is associative and during the course of a conversation people are often reminded of a previous experience, perhaps by a particular remark or an object. During the mental processing that is taking place during this conversation, some memory is involved to help understand the new input. As the new input reminds us of a previous experience, this suggests that we are using the same structure to process one experience and to remember the other. However the experience which is recalled may not appear to have any direct connection with the stimulus. Schank explains this phenomenon by considering memory as a series of interlinked packages (MOPs). Welbank (1983), quoting the example of asking people to describe the design on the reverse side of a coin, demonstrated that uncued recall of something that has never been specifically memorised is bad and that memory recognition is more complete and accurate than recall. For further study, Rummelhart and Norman (1983) provide an excellent study of how humans manage their own memory.

The history of the attempts at modelling mental states deliberately began with a top-down approach because it at least provided the AI research workers with a starting point. This approach has provided some success, but there is also a 'bottom-up' methodology based on nervous system structures. The work of Minsky and Papert (1969) discredited such early architectures, but with the provision of parallel processing machines there is a revived interest in a bottom-up methodology (eg neural networks and Boltzmann machines). This received further encouragement from Hopfield, reported by Durham (1987), who made the assumption that the connection between two neurons is symmetrical. The result being a content addressable memory or trainable pattern recognition device with mathematically analysable behaviour. While not making the production of neural networks any easier, this promised to make them easier to understand. The counter argument to this development is that, although these new machines may speed up computing and therefore do the present tasks much faster, they will never 'think' because scientists may never produce a complete understanding of what makes us think.

THE EXPERT SYSTEM

In chapter two, what constitutes an expert system was considered. It is the intention of this chapter to examine some of the characteristic features of expert systems.

One of the threads of the early work in AI was the attempt to develop models of the human brain. In the 1950s a move towards symbolic computing developed and one result of this move was the production of LISP. A further major landmark was the production of GENERAL PROBLEM SOLVER (GPS), (Newell and Simon 1963) a planning program which attempted to determine feasible sets of transformations which create a goal state from a given initial situation. It did this by repeatedly redefining the problem into a set of sub-problems and attempting to find solutions to these. It was able to succeed provided that it was applied to small, relatively simple problems. However, when applied to larger, more complex tasks, it failed because of the vast area of search space of possible alternatives to be considered. This 'combinatorial explosion' was identified by Lighthill (1973) as being a major restriction on the potential of AI research. Lighthill's investigation into the funding of British AI research concluded that

> "it was unlikely to bring much, if any, short term benefits and AI research was too costly and investment should be greatly reduced"

These conclusions were produced because of a possible misapprehension about what AI was trying to achieve and as a result of the over-exaggerated claims and failures of early work on robotics, vision and other systems. The research workers either did not appreciate, or did not show such appreciation in public, the complexities of the projects upon which they were working and Lighthill, among others, could not have been expected to foresee the speed of technological development that was to take place over the following decade.

Although GPS 'failed', it did produce some significant side-effects. GPS was too general, a fact observed by Feigenbaum (1979)

> "general problem solvers are too weak to be used as the basis for building high-performance systems. The behaviour of the best general problem solvers that we know, humans, is observed to be weak and shallow, except where the human problem solver is a specialist"

Based on this observation, Feigenbaum developed DENDRAL (Lederburg 1980), initially as a conventional algorithmic program. The importance was that it was a special purpose program intended to work in a narrow domain. This idea can be seen today in the development of highly specialised expert systems which work within limited domains.

One of the most important features of expert systems is the separation of knowledge from reasoning. In traditional BASIC or COBOL programs, the knowledge of the application AND the control or reasoning is hidden away in the program code. This, incidentally, means expensive overheads in the maintenance of traditional software. In expert systems, the knowledge of the application, often in the form of facts and rules (known as the knowledge base) is kept separate from the reasoning. The distinction between a knowledge base and a traditional database is that although they both contain structured information, the former also contains information about how to carry out the required task. It is held in an explicit form and thus the knowledge base can be adapted and amended independently from the reasoning or inference mechanism. The inference mechanism does not have to be 'application-specific' as does the knowledge base. For example the inference mechanism in a medical diagnostic system could easily find a place in diagnostic uses in engineering. However this does not mean that one inference mechanism will suffice for all problems.

The user interface, the third part of an expert system, allows the user to create and amend the knowledge base, to explore the knowledge base, (however some systems do not fully allow this, they require the knowledge to be developed and then compiled), and to consult the system.

Sviokla (1986) reported that 42% of the code in the DIPMETER ADVISOR system is dedicated to the user interface, with only 30% of the code making up the knowledge base and the reasoning mechanism. However, despite the smaller proportion of code, the latter is both more difficult, and will take more time, to create.

The user interface usually, but not necessarily, involves a human user interacting with the system in some form of 'conversation'. There are examples where there are no human users, as in real time control applications which accept input from external sensors and provide output for control devices.

One advantage of an expert system containing an explanation facility to explain its conclusions is that it may help in persuading the user to accept and understand the system's decision. An important factor with the use of an expert system is this ability to question the train of thought of the system. For example, if a doctor is uncertain why the system made a particular diagnosis, using the explanation facility, he can follow the steps taken to reach the particular conclusion. If a human senior consultant gives a decision with which the junior doctor does not agree or does not understand, then the junior doctor would have the option of questioning the consultant. For an expert system to have credibility, then similar features should be available to the user. The system designer has the responsibility for ensuring that the user and the system share the same vocabulary and understanding. Without this, any recommendations from the system could easily be misunderstood.

In addition the explanation feature will be of value in any subsequent debugging of the knowledge base and will also have an educational value which is discussed later. These explanation facilities can take various forms, this process could be a simple trace facility, but it is much more useful if more full explanation facilities are available. ES/P Advisor, for example, provides the following features

EXPLAIN - a facility which will provide an explanation of terms used during a dialogue, thus providing assistance for the non-expert user.

HOW ? - a facility to enable the user to ask the system to justify its conclusions. The system will backtrack to the previous stage of the argument that it has used and inform the user.

WHY ? - a facility enabling the user to ask the system to explain why it is asking a particular question. The system responds by explaining what goal or sub-goal the answer will help to prove.

Additionally it may be beneficial to allow the user to volunteer information without being asked for it by the system. This can save time in the execution of some situations, but this is dependent upon the application and in some situations it may be advisable to allow only that information which the system requests. Some systems do provide a mixed-initiative style interface, where the user can take control of the dialogue at any point. This can have considerable benefits because the resultant conclusion will have been derived from a combination of the expertise contained in the system and the knowledge of the user.

However, even if the explanation features are provided in natural language, where the dialogue takes place in English, or a sub-set of English, it is no more than window dressing if the system does not possess sufficient domain knowledge. It is possible that far more use would be made of the potential offered by computers in general, if they contained a true natural language interface. However this is a difficult research problem because of the complexities involved in computerising natural language, particularly as so much of natural language is context-dependent. Early failures, reported by Brain and Brain (1984), at producing automatic translators resulted in examples such as

> 'the spirit is willing, but the flesh is weak'

translated into Russian and back into English as

> 'the vodka is strong, but the meat is rotten'

> and 'out of sight, out of mind' translated as 'invisible maniac'

However natural language research is likely to produce the most useful medium for the man-machine interface in the distant future.

For a historical perspective and further analyses of the problems see Wallace (1984), Sparck Jones (1984), Sparck Jones and Wilks (1985), Hutchins (1986) and Johnson (1986).

Any explanation, whether in natural language or not, must be related to the circumstances at that moment in time. For example, as a custard pie wings its way through the air towards you, it would be inappropriate to be given an extensive explanation of its aerodynamic properties, its chemical constituents and even the recipe for its manufacture. A concise 'look out' would surely suffice !

In the end, it is the users of a system that will provide information on the success of the system. They may decide that the knowledge is inadequate and needs validating or that it is failing to deal with a wide enough range of applications and needs improving. Basden (1983) suggests that a successful application needs;

a) a blurring of the distinction between expert system and conventional computing techniques so that techniques are selected according to their usefulness

b) a good interface to other programs

c) a well-engineered man-machine interface

d) high run-time efficiency

e) availability on a range of machines

Furthermore, it will certainly be classified as failing if it lacks a suitable explanation facility. A good maxim for any system developer to bear in mind, when considering the man-machine interface, is never to underestimate the computer illiteracy of the end user.

Regardless of the specific characteristics of an expert system, of prime importance is the quality of the knowledge contained by the system. It is this aspect that is considered in the next chapter.

INFORMATION, DATA AND KNOWLEDGE

We can never have a full and complete knowledge of the real world and we have to continue as best we can with this incomplete knowledge and understanding. This situation is equally true when considering the use of 'intelligent' machines. Hence Brownowski (1973) sets the scene perfectly for this chapter

> "there is no absolute knowledge
> .... all information is imperfect"

Use of the word 'information' is accompanied by the danger of confusion between 'knowledge' and the image of knowledge provided by data. Indeed I have found myself using the terms synonymously during the research for this paper. There is a correspondence between knowledge and data, but they are not the same commodity. Knowledge is not a loose leaf folder full of facts. Bretz (1971) identified the distinction between information and knowledge as relating to structure.

> "Information has far less structure than knowledge: much information in fact consists of isolated and unrelated facts. In general, unrelated information can be filed in a human memory only when it has become associated with some prior structure of understanding and has become part of a person's knowledge."

Warnier (1986) provided the following distinction

> "Data may be viewed as the expression, in a certain language, of our perceptions of the surrounding world. Whenever knowledge is acquired an image of the world is constructed within ourselves"

Data is plain and unrelated, for example from measurements, estimates and observations. It lacks a context which can provide real meaning. Winograd and Flores (1986) noted that the actual meaning of information is not confined to the actual message, but meaning is constructed around the message depending upon the sender or recipient of the message. Information is structured data held in such a way that the relationships between data items can be identified and useful statements can be made.

The changing of raw data to information involves such processes as rearrangement, aggregation and correlation. It was noted by Raphael (1976) that once 'information' had a generally understood meaning, computer scientists redefined it as;

> "the amount of data that must be transmitted through a communications channel in order to convey a message, in all its detail, from one place to another."

Piaget proposed the concept of knowledge as a process rather than a state. In a similar vein, Newell and Simon (1972) described human problem solving in information processing terms where the behaviour takes the form of a sequential search, making additions to the information about the problem. Popper (1974) suggested that knowledge can be viewed as problem solving, by forming hypotheses about the external world and amending these hypotheses in the light of experience.

All living species survive by collecting information from their surroundings, processing, storing and translating it into actions that are aimed at facilitating their continued survival. In this respect Man is no different. Where Man is unique is in the degree to which this skill has been developed. Humans are processing information all the time, but in a selective fashion to prevent information overload, selecting from the mass of audio-visual signals bombarding us at every second to produce our perception of the world.

Information and knowledge are unique among resources in that they are not reduced or lessened by use or sharing. The information extracted from the environment by one organism does not reduce the amount of information available to other organisms. Similarly the amount that is learned by one does not reduce the amount that can be learned by another (see Davies 1969). Indeed sharing information can increase the amount of knowledge and learning that can take place.

If we take a look at our surroundings, we receive information about it. If we were to use a magnifying glass or a microscope for our observations, that information would be available in finer detail. That would have been the limit of observation a century ago, but more sophisticated observation tools are now available and the level of detail achieved by the electron microscope might have astonished our ancestors. Likewise the level of detail that may be available in a century's time will cause amendments and re-thinking of some of our current scientific theories.

A distinguishing feature of our species is the ability to imagine what is going on in other people's heads, including what they are thinking about us (this being a prerequisite of social interaction) and to juggle with various levels of meta-knowledge (knowledge about knowledge). However, as noted by Worden (1988)

> "meta-knowledge is almost never documented; in fact its intricate structure means that paper-based documentation would be rather ill-suited, and some form of computer-based support (eg Hypertext) may be necessary."

Knowledge is abstracted information enabling generalisations to be made by humans. It is this latter point that Dunn and Morgan (1987) stress

> "information only becomes knowledge when it is acquired and transformed by a person and used to make decisions etc."

For example, a doctor carries knowledge, obtained from various sources, around in his head and also uses information that is provided by, say, the chart on the end of the patient's bed. The quality of the use to which you can apply information relies on the quality of the presentation of the raw data. To make good use of information you need to relate the different elements and draw conclusions. Creative developments in many fields are often the result of someone combining together, in a novel form, a set of previously disparate pieces of information. There is an important distinction to be made between the possession of information or knowing how to find that information and the ability to use it, interpret it and present it. As noted by Dunn and Morgan (1987)

> "it is not the information - it is what you do with it"

The focus of computing attention is moving from data to knowledge. However this technological advance should not distract us, as educators, from a simple, yet plain, truth. Knowledge may be power, but the key to that knowledge is still reading and Weizenbaum (1984) warned of the danger of using the computer as a quick technological fix with this example

> "If Johnny can't read and some software will improve Johnny's reading score a little bit for the present, then the easiest thing to do is bring in the computer and sit Johnny down at it. This makes it unnecessary to ask why Johnny can't read."

The power of information within society was identified by McHale (1986)

> "the Watergate affair revolved entirely around who had obtained, or tried to obtain, which kinds of information, who transmitted what to whom, and when such information was associated with this or that power play."

In the UK, other affairs of a similar nature (eg the cases involving Clive Ponting or Peter Wright) lend further weight to McHale's identification.

Hence information has a valuable, if sometimes unquantified, history as an economic input. This is true up to the present day and will be even more so in the future. In the post-industrial economy, the number of information workers and users is set to expand. Stonier (1984) identified six categories of 'information operatives';

a) creators - scientists, artists, designers

b) transmitters - postal workers, journalists

c) storers/retrievers - librarians, filing clerks, computer programmers

d) appliers - doctors, lawyers

e) students - school, college,

f) organisation operatives - middle managers

The sort of knowledge which must be included in an effective expert system in a particular domain is dependent upon the activity within that domain. Knowledge as it is used in such systems falls into three basic types;

a) facts - the type of knowledge that is a simple description of the world
eg 'it is raining' is a fact

b) rules (or procedures) - the relationship between facts, what is used in manipulating and processing facts.
eg 'IF it is raining THEN use your umbrella' is a rule or procedure

c) control - the knowledge which determines which rules to apply in a given situation and how to cope with new situations

Cooley (1987) in discussing the integration of expert systems into engineering and manufacturing applications represents information and knowledge as areas between data and action.

```
         !
         !       * data
   ^     !
   !     !
         !         * information
 NOISE   !
   !     !
   !     !             * knowledge
         !
         !
         !                     * wisdom
         !
         !
         !                              * action
         !
         !___________________________________________
                SIGNAL -->
```

Klahr (1976) described that

> "if we attempt to represent knowledge in terms of networks by selecting labels for the concept nodes and relations, then although the result may be logically possible, it is an arbitrary

> schematic representation of knowledge that has little to do with how the concepts arose."

If an expert system is to be considered as skilled at some task, then it must not only contain an appropriate knowledge base, but also have the means to make effective and efficient use of that knowledge. Sridharan (1978) noted that one reason for undertaking the knowledge engineering exercise would be to codify valuable knowledge. Michie (1982b) has a table showing expert systems as sources of improved codifications of human knowledge.

Potential knowledge engineering applications occur anywhere where knowledge is not locally accessible, or is experiential, or demands the use of judgement. The following provides a list of possible examples of applications;

a) expertise is needed throughout an organisation and the expert is in only one location

b) current documentation is so bulky or

complex that people guess instead of using it

c) manuals and training courses need much revision because of frequent changes of rules, laws or methods

d) there is an excessive training requirement because of high staff turnover or frequent introduction of complex equipment

e) valuable enquiries are not dealt with because the only expert is too busy to cope

f) expensive labour is used for mundane tasks which only require a fraction of the expertise rather than on high value problem solving

g) there is a continual need to access and accumulate incomplete data

h) critical judgements have to be made in a very short time, to prevent expensive or disastrous situations developing

i) when not enough is known about a problem to build a deterministic model

j) making a good impression on clients

All of the above list are important commercial considerations, as will be discussed in Part Two of this thesis, but the first four examples would seem to be particularly relevant to educational applications and will be discussed further in Part Three.

KNOWLEDGE IS THE KEY

The power of expert systems comes not from any formal structure or inference mechanism that they may have, but from the knowledge that they possess. Hence they depend upon expert knowledge, because knowledge is the key ingredient in solving problems, whether we are discussing human or machine problem solving.

There must be sufficient domain knowledge present to enable solutions to be generated and there must also be heuristic knowledge to aid in reducing the processes of search. Human problem solvers become frustrated when there are too many potential and possible routes to the solution. Similarly, as noted by Michie (1986), a major part of AI research is avoiding or reducing the 'combinatorial explosion'. Banks (1986) noted that in many cases it is only possible to quantify the value of the knowledge, either in terms of its acquisition or its replacement when an expensive mistake has already been made.

There are two processes involved in putting the knowledge into an expert system, knowledge acquisition and knowledge engineering. The former is the process of procuring the knowledge, the latter is the process of coding it in the system. The term 'knowledge engineering' was coined by Feigenbaum (1980) after Michie's phrase 'epistemological engineering'. However, expert knowledge has characteristics which may make it difficult to be represented in a machine. There are three key issues involved with knowledge engineering;

a) comprehensibility (concerned with providing a system in which the end user can understand what is going on)

b) debugging (what happens when two experts disagree)

c) elicitation (getting the knowledge in the first place and then modifying or adding to it later on).

This whole process is seen as a bottleneck because it is often a lengthy, complex procedure which is both labour intensive and error prone. The knowledge is usually built up as a result of consultations between the system builders and the domain expert(s). The expert should be cooperative, communicative and suitably motivated (why should I give up all my expertise ?), but should neither be seen as a passive fount of knowledge.

A knowledge engineer is the person who will be most closely involved with the development of the expert system, but it doesn't have to be an individual, it may be better to think of it as a 'middle-man' role that needs to be performed to ensure the success of the project. Hayes-Roth et al (1983) observed that

> "One of the most difficult aspects of the knowledge engineer's task is helping the expert to structure the domain knowledge, to identify and formalize the domain concepts"

The qualities of a good knowledge engineer include an analytical and logical approach, an understanding of the tools and techniques which can be applied, an ability to understand, but not necessarily become an expert in, the problem domain and a high level of inter-personal skills.

Some systems have been developed that can induce the rule-base from given examples (eg Expert Ease, a system developed for the 128K IBM PC and the Apricot and Super Expert, an improved version). The technique of knowledge induction is discussed in more detail in chapter ten.

As the knowledge is teased from the experts, the knowledge engineer must ensure that everything is covered, suitable defaults are set and internal contradictions are sorted out. This is particularly true as the size of the knowledge base increases and so a scheme must be incorporated to prevent the addition of contradictory items of knowledge or to allow rules to be adapted in the light of the new knowledge. For example, if we have a rule

IF X is a bird THEN X can fly

It is obviously satisfactory if either

X = a sparrow

X = a kestrel

but we would have a problem to resolve if

X = an ostrich.

Similarly, if there are the following three rules

(1) IF X THEN Y

(2) IF Y THEN Z

(3) IF Z THEN X

and we need to evaluate X, then rule (3) would be used, which would call rule (2) to find Z. This would entail using rule (1) to find Y. As the value of X is required by rule (1) and X is already being evaluated by rule (3), a circular situation has been achieved. Rule (1) is not necessarily faulty and might, indeed, be a significant rule, but some mechanism must be included to extract the system from this circular reasoning.

Once the expertise has been written down, it may be readily refined. It was reported by Hewett and Sasson (1985) that Aldo Cimino's 44 years of experience with Campbell's Soup had been encapsulated into a few hundred rules. Chapter thirty two discusses whether the ability to see a man's experience laid out on a couple of sheets of paper can be considered as progress.

Additionally, because people turn out to be very good at improving a well-expressed heuristic, the knowledge engineering process may convey fresh insights not only to trainees, but also to experienced executives. This may provide another educational application of expert systems which will be considered in chapter twenty seven.

The expertise contained within a system is crucial for the initial success of that system. The capacity for adaptability and improvement is crucial for its continued success as users of the system usually adapt or fine tune the system as a result of their experiences with the system. However, new knowledge (a better system) comes from an expert, induced knowledge will only provide new links between existing knowledge.

Once the knowledge has been elicited it will require refinement and checking. In fact the knowledge base should be considered as a live system which needs to be continually kept up to date and it may never reach a state of being entirely complete or correct in any absolute sense. Sell (1984) observed that

> "Expert systems, like works of art are never finished, merely abandoned"

KNOWLEDGE REPRESENTATION

Knowledge representation is concerned with the problem of how to represent knowledge in a form which the computer can 'understand', so that the system can act in an 'intelligent' manner. This chapter looks at a selection of the techniques of knowledge representation.

Levesque (1984), along with others, noted that the knowledge base is one of the fundamental components of an expert system and the concept of knowledge representation is fundamental to the understanding of expert systems. It is not only the substantive content of the knowledge which determines the usefulness of the system in problem solving, but also the form in which that knowledge is made available. A good representation scheme should;

a) facilitate computation

b) make the important things explicit and suppress unnecessary detail and expose any constraints

c) be complete, in the sense that it should be capable of representing everything that needs to be represented

d) be transparent so that what has taken place during the dialogue can be understood

This provision of a good representation scheme is not a trivial problem. Our number systems can be used as an example of the need for a 'good' representation. Alty (1985b) identified the fact that the Romans were prisoners of their numeral representation system. Roman numerals were perfectly adequate for measuring quantities, but became useless when computation was required and consequently made no significant contribution to the development of mathematics. Base ten arabic numbers are far more useful in this respect.

In representing knowledge for use in computer systems, it is useful to use the concept of objects, which often have illustrative attributes, and the relationships between the objects.

The football (Object)	belongs_to (Relationship)	Neil (Object)

In some systems, for example MYCIN (Shortliffe 1976), the attributes have assigned values, creating what are known as associative triples.

The football is white	
Object: the football	) an
Attribute: colour	) associative
Value: white	) triple

There is no one way to represent knowledge, no universal formalism, different problems require different representations and there is little psychological evidence that humans use a single representation scheme for encoding information. Real world knowledge demands a generality which was not available on any of the early representation schemes. This restriction has meant that expert systems have had to be domain specific.

The following are examples of various representation schemes.

Logic

The study of logic is one of the foundations of AI and no study of expert systems would be complete without reference to it. The representation of problem domains is such that it is equivalent to the development of a valid argument, in which conclusions are drawn from a set of assumed facts. Logic generates the confidence in these arguments. The principle branch of logic which is concerned with expert systems is that of predicate calculus. A predicate is a logical function which operates on logical variables or arguments and the structures are assigned a truth value of either 'true' or 'false'. Logic is used as a representation and, in some ways, as an inference mechanism. It is possible to generate, in an algorithmic fashion, a proof of the proposition in terms of a set of assumptions. Indeed it is the existence of such algorithms which are the basis of PROLOG, a language based on predicate calculus as described by Barr and Feigenbaum (1981) and Alty and Coombs (1984).

Semantic networks

As semantic networks are relational, they are a particularly useful method of representing complex factual knowledge. Semantic nets were used in the PROSPECTOR system (Barr and Feigenbaum 1981) to represent knowledge in the domain of geology. They are based on the idea that memory is formed of associations (arcs) between concepts (nodes) and one focus of research in this area (Sathi, Fox and Greenberg 1985) is to identify standard types of nodes and arcs. A further attraction of this type of representation is that it is only one step away from natural language.

They are frequently shown as diagrams

```
                      is_a
Tottenham ----------------------> football club
Hotspur

                play_football_at
Tottenham ----------------------> White Hart Lane
Hotspur
```

One of the advantages of using this notation is the inheritance characteristic of the 'is_a' type relationship. Objects which belong to a certain class, or have a certain attribute, can inherit an indefinite number of other attributes and relationships. Inheritance would not be possible using associative triples.

Since their introduction in the late 1960s, sophisticated improvements have been developed, in particular the concepts of strictly limiting the types of links and nodes allowed and of partitioning off sections of the network. Hayes (1977) noted that one disadvantage is that they are passive structures and need an operator, which needs to be more complex than a rule-based inference engine to manipulate them, as shown by Quillian (1968) and Brachman (1979). For an analysis of the problems of building networks with sufficient expressive power see Brachman (1977) and Schubert (1976). A further detailed analysis of networks is provided by Nilsson (1982).

Frames

Frames are relative newcomers and were not used in any of the 'classic' expert systems. The notation of frames is similar to that of a traditional computer database record in that it has a name to which is attached a number of labelled slots which can contain the name of another frame, a constant value or 'compute' values which enable frames to control numeric processing where appropriate. They also have the useful property of allowing defaults. Hence they can represent hierarchical characteristics enabling the system to learn about itself. Amongst the problems and limitations is included the fact that it may be difficult to match frames to the elements of a problem. Systems based on frames tend to require expensive hardware and run slowly because of the complexity of the interactions in large frame-based systems. Finally, developers cannot usually anticipate run-time behaviour.

Minsky (1975) and Charniak (1978) showed that the declarative nature of production rules and frames enables the representation within the same knowledge base of both the static structure of the system and rules relating components. Minsky (1975) contends that knowledge must be highly structured and suggested an approach of incorporating a network structure with a frame system. Indeed over the years, the distinction between frames and networks has become blurred.

Scripts and plans

Schank (1975) provides a detailed summary of scripts and plans, which can be viewed as a kind of semantic network. They are meant to describe situations where the events and participants are stereotyped, so that they can be expected to happen every time the particular situation is encountered. For example if 'Neil' and 'football' are present, then we arrive at the idea of 'Neil' kicking the 'football'.

Production rules

The fundamental element of a production rule is the condition-action pair, for example

IF x THEN y

or IF a AND IF b THEN c

This is a powerful method which can fairly quickly generate an efficient, well understood solution. However, the designer must have a clear understanding of the solution method and the domain will produce a narrow system. The rigid syntax does afford other advantages in that consistency checking is quite readily incorporated and it leads itself to the easy production of explanations.

Although production rules are easy to formulate and input, in order to construct an effective knowledge base, it is necessary to be aware of the structural links between the various rules. The technique of initially constructing the rules as a tree diagram has the advantage of focussing attention on one branch at a time and thereby breaking the problem down into smaller sections.

The representation of an expert's knowledge may consist of many rules and there may well be situations where more than one rule may fire. A system of conflict resolution is needed at this point. Similarly, there may be within the knowledge base, a number of rules variously proving and disproving the conclusion. In practice, though, such systems are more likely to include rules which prove rather than disprove things.

Production systems are an established, though not universally accepted, method of modelling human cognition. Human cognitive processes are executed in a system which comprises both short-term and long-term memory. The short-term, or working, memory contains between 5 and 9 'chunks' of information (Miller 1956). The rules reside in long-term memory. The system works by recognising the contents of the working memory and carrying out the respective actions. The activity of a production system consists of a sequence of rules being fired in response to conditions occurring in working memory. In the context of expert systems, the knowledge base corresponds to the long-term memory and the inference mechanism occupies the short-term memory.

While production systems may provide an appropriate representation in some domains, this is not universal as demonstrated by Davis and King (1976) who challenge the precept that production systems function in the same way as human cognitive processes. Production rules can only work through the knowledge base in relatively small steps and are best suited to domains where the knowledge can be divided into small sections. Alvey and Greaves (1986) noted that, particularly in operational, rather than demonstrator, systems, good coordination between rules was vital, although, Newell and Simon (1972) noted that meta-rules (rules about rules) can be employed to increase the power of the systems.

Blackboard

A blackboard system is a set of expert system building tools rather than a frame for storing rules and data. It can be likened to human brainstorming sessions where the knowledge is stored in 'knowledge sources' (visualised as a number of domain experts scribbling their contributions to the 'whole' on a blackboard) and these communicate with each other via a common data structure known as a blackboard. They have several advantages (as do brainstorming sessions by humans) in that being highly parallel, they allow the opportunistic approach to problem solving and allow proper separation of the available knowledge in various domains. Engelmore and Morgan (1988) provide descriptions of a number of developed blackboard systems.

A further advantage that may become apparent in the course of time is that blackboard architecture allows for the easy implementation of object oriented programming features, the most relevant of these at this moment is the ability to distinguish classes as abstract data types on the blackboard.

Since the early 1980s there has been the development of environments which support a variety of knowledge formalisms. For example, Knowledge Craft (developed at Carnegie-Mellon University) combines a schema-based representation of data with rule-based programming. Other examples include;

Inference ART - Ferranti
KEE - Intellicorp
LOOPS (Stefik et al 1983)

Studies such as McDermott (1984) and Smith (1984) provide an analysis of the relative strengths and weaknesses of the various architectural options. Appendix 1 provides further examples of inference mechanisms or reasoning styles.

But how can we represent common sense ?

This is a form of knowledge that covers a broad spectrum of worldly general knowledge including self-knowledge (knowledge about what you do and do not know) that everyone possesses to some degree. However, the mere possession of it is no guarantee that it will be used. The sheer size of it though makes it difficult to include in any expert system. For example, human common sense would quickly detect several errors in the following sets of data

	height	age
John Smith	125 metres	10 years
Alan Smith	35 cm.	180 years

In the case of John Smith, the units of height are probably incorrect and in the second case, the units are correct but the 35 and 180 have probably been transposed. Unless a check feature, such as a look-up table, had been incorporated into the system, the crucial question is whether the expert system would accept the error in the data. Weizenbaum (1966) has shown that many expert systems can be made to appear to act in a stupid way.

The expertise of humans degrades gently as they approach the limits of their expertise. In contrast, expert systems tend to reach a brick wall. By using an expert system as an aid, the human user can supply any missing common sense. Buchanan (1986) observed that MYCIN could accept the possibility of pregnancy for males, but the doctor using the system would surely have made the necessary allowances. The important point to note is that the more relevant knowledge that an expert system has, whether common sense or domain specific, the more likely it is that the system will perform well.

DEEP AND SHALLOW

The concept of deep and shallow can be applied to three aspects of expert systems; the knowledge involved, the reasoning strategy and the domain of the application.

Deep and surface knowledge

Although deep knowledge is more concise and abstract than shallow knowledge, there is no hard and fast boundary between these two types of knowledge; a continuum relating to 'depth' of knowledge may be a more useful representation.

Worden (1988) argued that the starting point of the 'knowledge life cycle' is from uniformly shallow knowledge and as the field of knowledge develops, more and more deep knowledge is introduced and this evolution from shallow to deep is often accompanied by an evolution from procedural to declarative. In general, 'declarative knowledge' is that which makes statements of what is the case and 'procedural knowledge' is that which makes statements of how to do something. In addition, it is necessary to relate the use of the term 'knowledge' to the level of application. It is possible to refer to knowledge about the world, about the facts, about the representations of the facts and so on, the scope depending upon the level of consideration. DEPAM (Sriram and Rychener 1986) is an example of a system which attempts to combine shallow and deep knowledge.

The incorporation of 'deep' rather than 'shallow' knowledge into expert systems has implications for tutoring systems. The latter is very specific to a particular application, whereas the former is concerned with more general principles. For example, in MYCIN the procedural and declarative knowledge were mixed in the same rules. These could not be used for teaching until they were more clearly separated in NEOMYCIN.

Deep and shallow reasoning

First generation expert systems have been criticised for using shallow reasoning. Steels (1986) noted that a number of problems with current systems could be overcome by combining rule-based heuristic reasoning with deep reasoning, based on a model of the problem domain. Hence such second generation systems will have deep and shallow reasoning combined. Diagnostic systems have a functional model (essentially a simulation of the system under study). Shallow reasoning systems give results faster, whereas deep reasoning systems, based on causal models (those which embody more abstract knowledge of cause and effect), require a lot of computer time as they consider many possibilities and may not provide a solution. Many researchers have seen machine learning as a way of deriving rules from examples or refining rules into more compact statements. This use of learning is refining shallow knowledge into deeper knowledge. Steels, reported by Durham (1985), suggests that the expert system should start with a body of deep knowledge, but should accumulate shallow or heuristic knowledge from experience. In other words, it should learn short cuts which have worked before.

Three types of knowledge are used in expert systems;

a) deep knowledge, the formal theoretical learning that contains verifiable facts and theorems, definitions which may be independent of the domain area

b) heuristics, the rule of thumb knowledge that is acquired by problem solving behaviour before learning the underlying expertise

c) empirical or compiled knowledge, organised in such a way that it is easily accessible for problem solving (eg a car repair manual). This can be as deep knowledge or heuristics stored in modules of the knowledge base.

Shallow reasoning operates on empirical knowledge, whereas deep reasoning requires more sophisticated knowledge representation techniques and inference mechanisms. Systems which only apply shallow reasoning are not acting as experts, who by definition have an understanding of the underlying theories. For example, a shallow fault diagnosis system is unlikely to be able to help if the specific fault and the associated symptoms do not appear in the knowledge base. However it will perform a useful function provided that it is used within a limited domain. A system capable of a deeper level of reasoning will, on not finding the specific fault, be able to fall back on theory and basic principles. It will therefore be a more useful tool and will emulate, to a greater extent, expert human behaviour. CASNET (Barr and Feigenbaum 1982) and ESCORT (Turner 1986) are examples of systems which attempt to employ deep reasoning strategies.

Deep and narrow domains

Science and engineering are examples of deep and narrow domains which are very limited in scope and domain expertise is gained by deeper rather than wider understanding. Wide and shallow domains are wide ranging in scope (eg economics) and there is often little agreement about the relative importance of the various factors. Naughton (1986) made the point that

> "the ability to spot the essential commonality between two situations is something which humans have, or can be trained to acquire. Embodying such abilities in machines is difficult, but essential if expert systems are to become more 'generalisable' across domains."

KNOWLEDGE ACQUISITION

The gaining of knowledge is not an easy occupation and in a quotation attributed to Somerset Maugham, methods of easing the acquisition of knowledge were considered.

> "It is a great nuisance that knowledge can only be acquired by hard work. It would be fine if we could swallow the powder of profitable information made palatable by the jam of fiction"

Knowledge acquisition or elicitation is the process of gathering the expert knowledge before entering it into an expert system. This is a lengthy process as the information is extracted from the experts, checking that everything has been covered, internal contradictions have been settled and suitable defaults have been established. It turns out that people, in general, use a great deal of implicit, unstated knowledge. Attempting to formalise this knowledge is a challenging task which is further complicated when the knowledge acquisition may come from several sources and is from several experts. Under such circumstances, the knowledge may well be contradictory or inconsistent and some means of checking for such cases must be included by the knowledge engineer.

However the actual acquisition of knowledge is not the end of the matter, as back in the garden of Eden, Adam and Eve discovered that the acquisition of knowledge wasn't always fruitful !

The methods of knowledge acquisition can be grouped into three.

Being told

This is the simplest form of knowledge acquisition, the knowledge (information about the domain and how to use the facts) is entered into the knowledge base and can be checked, used and amended by the knowledge engineer. However, it would be beneficial if the system could perform its own checking of new knowledge. For example, when a new item is entered, it may;

a) already exist, either explicitly or by implication, within the knowledge base. If this is the case then the item should, on the grounds of efficiency, be rejected.

b) be inconsistent with existing knowledge. If this is the case, either the new item should be rejected or the knowledge base amended to allow consistency with the new item.

c) be a new item, in that it is not already contained in the knowledge base, neither is it deducible from that knowledge nor is it inconsistent with the existing knowledge base. In this case, the item should be incorporated into the knowledge base, but once it has been added, the system needs to check if there are now any redundant rules as a result of the latest inclusion.

There may be theoretical or practical problems when deleting items and it was proposed by Kowalski and Sergot (1985) that rather than deleting an item, it should be marked as being no longer 'true'. Hewett, Timms and d'Aumale (1986) report that the XCON maintenance team have found that it may be more problematic to delete rules that have become redundant rather than leave them. They report that adding and deleting rules to any knowledge base can have unpredictable side effects. In practice the idea of never deleting an item would need vast memory storage capacity and though this may happen in time, for all practical purposes it has only limited validity.

Induction from examples

An expert often finds difficulty in explaining and making explicit his implicit knowledge. However, the expert can often provide plenty of examples about the task he performs. The major problem with induction is not just the need to discover the underlying pattern in a series of examples, but to extrapolate the knowledge to deal with new events and to ensure that sensible and useful or interesting consequences are the result of the induced knowledge. Cohen and Feigenbaum (1983) reported that AM made a series of useful and 'new' discoveries in the domain of symbolic maths, but when the program was applied to numeric maths, an area not contained within the program, it made many 'uninteresting' conjectures. Knowledge acquisition is often quoted as a problem (eg Hayes-Roth, Waterman and Lenat 1983), but it has been suggested by Michie and Johnston (1985), that in the future inductive systems may be useful sources of knowledge.

Observation and discovery

Many current systems do not significantly learn and improve their performance as a result of experience. It is a criticism of expert systems in general that until they are capable of doing so, they cannot be considered to be acting in an 'intelligent' fashion. See chapter thirty for an analysis of machine learning.

Knowledge acquisition is not a trivial phase as the experts are not always fully aware of the nature of their expertise. An example is the story, quoted in Michie (1982a), of the quality control department of a French cheese factory. This department consisted of one very old man who simply pushed his finger into the cheese and pronounced it good or not. The company, appreciating the old man's age, wished to computerise his expertise so that it would be available after his death. The knowledge engineers interviewed the old man at length in an attempt to produce some form of 'cheese testing expert system'. They delved into such ideas as the weight of the cheese, the thickness of the cheese crust and even the force used by the finger. On subsequent examination it was found that the old man's skill depended not upon any tactile sense, but upon his sense of smell, in that he poked his finger into the cheese and detected the quality from the aroma that was released. The important point here is that the old man was not aware of his skill and this is true of many experts in that they perform their tasks without fully understanding how they function. A further complication is that the more skilled the 'expert' becomes, the more difficult it may be to explain the reasoning process to anyone else. The nature of expertise is such that many skills (eg walking) are performed on 'automatic pilot'. For example there may well be problems if you had to consciously think about the necessary actions required for walking.

Although the acquisition of knowledge has been seen as one of the hardest aspects of building expert systems, Young (1984) argued that the problem lay not in the acquisition, but in the weakness of the available representation techniques, as was discussed in chapter nine.

<u>Acquisition techniques</u>

Boose (1985) provides a structured approach to the process of knowledge acquisition. Dawkins (1986) identified three stages in the process of knowledge acquisition, the elicitation, the analysis of data and the actual input to the knowledge base. There are some tools available to carry out the elicitation stage. For example, TEIRESIAS (Barr and Feigenbaum 1982) is an 'intelligent' editing program whose aim was to reduce the role of the system builder by allowing the expert to interact directly with the system.

A number of elicitation techniques can be employed, Buchanan (1982) and Smith and Baker (1983), discuss introspection, a technique which relies upon the expert to act as the system builder, identifying the basis of his knowledge and incorporating it into the system. This conceptualisation may be difficult, as the experts may not be able to describe their expertise in a rational and structured form.

The need when employing the 'observation and thinking aloud technique' is for the interviewer to watch the expert solving problems in the course of his usual work, without interruption. The importance of not interrupting is that the expert may feel pressured into providing a line of reasoning that suffices at that moment of time, but is not representative of his usual reasoning processes. At a later stage the interviewer analyses his notes and transcripts in an attempt to identify key concepts and relationships. The major problem with this method is that it is very easy to build an incomplete knowledge base and the gaps could prove difficult to fill at a later stage.

Interviews, involving a mixture of introspection, observation and interrogation have become the most frequently used method. Bainbridge (1981) showed that it was a faster technique than observing and Myers et al (1983) coded directly from a text editor to rules to produce a quick prototype to use to supplement interviews. This technique combines the problems associated with the other techniques, but it has the advantage that it can allow intervention by the system builder and introspection by the expert. Although it has become a widely used technique, Gammack and Young (1984) have noted that not all knowledge can be elicited by interview and Johnston (1985b) reported that during a formal interview it is easy to become removed from reality. A further disadvantage was noted by Calderhead (1984) in that when reflecting upon events, there is a tendency to rationalise decisions which had been made.

A different approach involves the expert describing WHAT it is he does and not HOW he does it. These examples are then fed into an induction system which induces general principles. The advantage of this method is that it is often easier for the expert to provide examples of decisions rather than describe the actual decision making process. The disadvantage is that the examples fed into the system must be selected carefully to cover all potential eventualities.

Using the 'model critique technique', the system builder constructs what he believes to be an accurate model of the expertise of the expert. This model is then criticised by the expert and refined in the light of the expert's comments. The system builder needs to achieve a balanced model, not too trivial and not in a rigidly preconceived form, but which approaches the expertise of the expert. To get this balance right is itself not easy and for this reason, this method is considered ineffective, unless combined with other techniques or unless the system builder has a sufficient level of domain understanding.

A variety of psychological approaches have been attempted, for example the Construct Theory of Kelly (1955), further described by Shaw (1981) and Hart (1986), identifies the links between various domain concepts. Eliciting a concept hierarchy, relative to the specific domain, from the expert was suggested by Chi, Feltovich and Glaser (1981) to be particularly applicable to classificatory knowledge.

Protocol analysis, as described by Newell and Simon (1972) and Ericcson and Simon (1984), can go beyond the information that the expert can explicitly provide and this may be of particular importance in problem-solving areas. Closely associated with protocol analysis is an analysis of the task under examination. The result from the two analyses, taken together, could provide a greater insight into the domain knowledge.

Gammack and Young (1984) attempted to match the type of knowledge with the type of knowledge acquisition techniques. As the situation of each expert and his expertise will be different and as the domain knowledge is also different, it is not possible to give an ideal prescriptive method of knowledge acquisition, other than to employ a balanced approach, combining the appropriate strengths of the above techniques. Whatever technique is finally selected, the process should be iterative. Once enough knowledge is available to build a system, it should be constructed, as that prototype system could then be used to aid further knowledge acquisition by identifying possible gaps, patterns or mistakes in the existing knowledge. However patience is required at this phase as Dyer (1988) suggested that

> "knowledge acquisition should not be regarded as a marathon to be finished as quickly as possible, or as a road block that must be gotten around or over before the real program development can begin. It is an information exchange process."

Friedland (1981) stated that expert knowledge comes in two forms, a declarative form, (the acquisition and representation of which is well documented) and procedural, which remains a major research issue. The paper (in the field of molecular biology) defined four classes of procedural knowledge;

a) data manipulation - the rules which ensure completeness and consistency

b) simulation procedures - the information needed to alter the representation of objects

c) selection heuristics - used to choose between alternative options

d) experiment design strategies

If it is agreed that it is desirable for a knowledge base to contain the above types of knowledge, then the traditional approach would be for a knowledge engineer to codify the knowledge into the expert system. In the MOLGEN project (Cohen and Feigenbaum 1983), the emphasis was on the domain experts building the knowledge base themselves. The thinking behind this approach was that accuracy and completeness of the knowledge base, particularly the complex and subtle aspects of the knowledge, suffers when it passes through a chain or filter of domain 'non-experts'. Furthermore a large knowledge base may be built in a shorter time and an element of trust is implicit if it is known that the knowledge base was constructed by a respected domain expert.

Any knowledge elicitation technique or combination of techniques must get both good and bad information and this can show up the limitations of an expert's knowledge. The situation can be improved by rapid prototyping so that the knowledge engineering process becomes almost interactive. Hence the knowledge engineer should get deeply involved with the domain expertise.

Johnston (1985b) reported how much of an expert system can be built without an expert by relying on theoretical causal knowledge of how systems ought to work, to be adjusted later in consultation with operational experts. They found that no one technique of knowledge elicitation can achieve success and what is needed is a combination. A 'common sense' approach allied to a detailed training period, was employed, which had the advantage that it was possible to build a system without all the problems of knowledge elicitation, but with the disadvantage that the process would take a long time and that the learning may not be successful. The conclusion was that it is possible to build a system without an expert but the domain had to be carefully chosen and there should be a domain expert available for tuning the system.

Similarly, Thompson and Clancey (1986) reported that in the development of CASTER, the system builders attempted to become their own domain experts by referring to an expert only when the standard sandcasting textbook failed to provide the necessary help.

A different approach was adopted by metallurgists at Westinghouse Electric Corporation (Expert Systems 1986). They, the domain experts, developed a system themselves rather than using knowledge engineers. This meant three months learning LISP and other AI techniques, but they felt that this was a better approach than trying to train a knowledge engineer in the basics and not-so-basics of metallurgy. Another example is provided by Ian Taig, also a domain expert, who constructed FEASA without a knowledge engineer.

The traditional extraction of knowledge from the expert has proved slow and difficult, machine learning techniques may be able to automate part of the knowledge elicitation process including the 'prioritisation' of rules, internal checking for consistency, prompting for new information and integrating the new knowledge into the knowledge base. These techniques are further discussed in chapter thirty.

Experts may have doubts about the role of expert systems and consider whether they may be doing themselves out of a job. However, Berry and Broadbent (1986) report that this doesn't seem to be the case, as existing systems tend to enhance rather than replace experts. Likewise, Basden (1983) doesn't think that expert systems will supplant human experts, the latter are likely to always have greater expertise in many domains. Even if expert systems could 'learn', human experts would still have the advantage of being able to recognise and take account of extraneous factors such as economic or political considerations.

Knowledge acquisition has been claimed to sharpen up an expert's thinking and generally to be very enlightening. In a medical application, reported by Brown (1985), it was found that the doctors who were being used as the experts were very interested in finding out how they do what they do. This is achieved by forcing the expert to articulate every step of the judgement process, something which in the normal day-to-day working routine would be eliminated. Experts have a tendency to state their reasoning and conclusions at a high level. This careful reflection and reconsideration of the expert's own reasoning processes was found by Kidd and Welbank (1983) and Klahr and Waterman (1986) to make the expert more critical of his own methods. If this is true, then there may well be an important educational value in this approach.

KNOWLEDGE INDUCTION

The process of knowledge acquisition has been seen as a bottleneck, particularly where the number of experts is limited. Automatic knowledge induction systems can be useful in situations where there is time pressure and the system builder may not have sufficient time to consider all the relevant data. This involves feeding examples into a logical induction algorithm such as ID-3 (Quinlan 1979b). However, Rauch-Hindin (1986) views these induction systems as being at an immature stage and inadequate for many applications.

> "Systems can acquire enough knowledge to make them very competent, but if an unusual situation arises such as a potato in the exhaust pipe, then no matter how good the prior performance of the system, here it will fail and have to seek help from the human.

> The human expert will then add rules enabling the system to diagnose a potato in the exhaust pipe. Everything will then be fine until the car breaks down with a turnip in the carburettor and it will be back to the expert again."

Automatic knowledge acquisition methods infer details and possibly gross structure from a given set of examples. This may result in the production of a set of rules based on the examples, but it is not the same as machine learning, which will be discussed in chapter thirty.

Human expertise is a mixture of observations, context, problem solving strategy and understanding. Induction techniques attempt to produce rules which are only based on the first two components and with no understanding or problem solving strategy.

Mitchell (1982) provides a compact representation of all the inductive hypotheses that are compatible with a training set of both examples and non-examples. The basis of his idea is to produce a continuum of admissable hypotheses by storing the most general hypotheses that do not imply any non-example and the most specific hypotheses which do not exclude any example. This agrees with the approach of Lenat (1983)

> "if a heuristic is occasionally useful but usually bad, then add specialisations of the heuristic"

Michalski and Chilausky (1980) and Hart (1987) argued that induced results will be good only if a good inductive algorithm is used on a training set which contains adequate information, in a suitable form, about the problem. The training set must be 'good' and not just a random selection of examples. This is similar to the same way that a good teacher selects representative learning examples.

Michalski and Chilausky (1980) showed that inductively produced rules performed better than rules derived from experts, although the source of the examples was not made clear. It should be noted that the function of explanation is different from the function of diagnosis, so that experts in making diagnoses are not necessarily experts in explaining their process of diagnosis. If this is the case, then the reliability of the data describing diagnoses made by experts (ie the reliability of the learning events) will tend to be better than the diagnostic decision rules which they formulate (does this provide another argument for knowledge acquisition by induction ?) The paper concluded that current (1980) induction techniques can already offer a viable method of knowledge acquisition if the problem domain is sufficiently simple and well-defined.

An interesting advantage of induction techniques was reported by Hewett, Timms and d'Aumale (1986) in that the use of Expert

Ease had identified 4 process measurements, out of a total of 25 that were routinely taken, that were actually relevant to a particular analysis at a nuclear processing plant.

Induction techniques may prove to become a routine tool in many domains, but the rules that are induced must still be checked by a human expert, as the expert system has no understanding of the rules that it has induced or the rules on the periphery of the rule-base. Suwa et al (1982) noted that there is a problem of checking the consistency of induced rules as many inconsistencies are very subtle and even the knowledge engineer is not likely to spot them. As regards the educational applications of expert systems, a further, and perhaps more serious, problem with inductions from examples is that it is a method that does not produce automatic explanations.

UNCERTAIN KNOWLEDGE

In many, if not all applications, the knowledge may not be completely certain. There are several reasons for this, as the user may be uncertain because of;

a) the relevance of one piece of knowledge to another

b) the truth or otherwise of a piece of knowledge

c) the likelihood of a simple, compound or conditional event

d) the incompleteness of the information

e) the imprecise nature of some knowledge whose behaviour obeys laws of statistical distribution rather than absolute laws

f) 'noisy' data

g) the knowledge where categories cannot be quantified and/or where relations are expressed qualitatively

h) any exception to a general rule

In short, the information wanted may be vague, it may be missing or it may be wrong.

Reichgelt and van Harmelen (1985) refer to uncertain terminology, where one term may mean different things to different people (eg does 'frame' refer to computer, window, garden or snooker ?). This issue, which I note as being important, is outside the scope of my work.

Dealing with uncertainty covers two distinct but related issues, how to represent the uncertainty and how to make decisions in spite of it. The problem of distinguishing between uncertainty with complete information from a lack of information was solved in HEXSCON (Wright et al 1986) by inferencing belief and confidence parameters.

Many problems tackled by expert systems involve a degree of information uncertainty. One solution is to allow default reasoning where certain default values are maintained unless specific contrary information is received. In a small system, it may incorporate a 'guessing' facility to provide a 'guestimate' of the missing data. The problem with this method is that any inaccuracies will be magnified through successive inference steps and therefore as the size of the model increases the degree of accuracy will decrease.

The difficulty of saying anything meaningful about a system was noted by Bandler and Kohutt (1980) who showed that it increases enormously with its complexity. In any real world situation our information is too voluminous and intricate and needs to be summarised or it risks being approximate from the beginning. Models of the system may be built, but any unwarranted structural assumptions imposed on the working model can lead to meaningless results. When data is uncertain, then the accuracy of lower parts of the decision tree need investigation. Hart (1986) suggested growing a tree to the full and then pruning it back, based on trading off the cost of a more complex tree against the risk of misclassification. It was shown that this usually gives better results than stopping growth during induction.

Some systems provide for facts or rules with associated probabilities or certainty factors. The problems addressed by such systems are largely those of indefinite knowledge and imprecise data. The handling of unreliable data and knowledge is achieved through some form of weighted evaluation. However, these weightings can be biased, particularly where value judgements are involved. Shweder (1977) suggested that experts overestimated the positive weights of evidence and Kidd and Cooper (1983) reported that experts would specify and rank probable faults, but would not give numerical values for the probabilities.

The techniques for handling uncertain data include fuzzy logic (Zadeh 1979), Bayes theorem, as used in PROSPECTOR (Duda and Gaschnig 1978), and certainty factors, as used in MYCIN (Shortliffe 1976).

Fuzzy logic

In fuzzy logic, it is important to distinguish between 'fuzziness' and just vagueness. Fuzzy logic measures the truth of a statement as a number between 0 and 1 and may be regarded as a probability factor. The underlying concept is that of 'partial' membership of a set. For example, if asked to "give me a large number", where 'x is the large number'. It can be represented by a set of likelihoods of 'x' being a member of several sets. Depending upon who is asked to provide the large number, then the possible likelihoods could be

x < 10	0.1	likelihood
x >=10 AND <100	0.2	"
x >=100 AND <1000	0.5	"
x >=1000	0.2	"

Probability factors can be considered as combinations, for example;

A is true with a value of 0.9
B is true with a value of 0.4

Using a rule such as

IF A AND B THEN C

then the minimum probability for all antecedents is taken (ie 0.4)

IF A OR B THEN C

then the maximum probability for all antecedents is taken (ie 0.9)

<u>Bayes Theorem</u>

Reverend Thomas Bayes, in the early 1700s, considered how worldly evidence could be used to prove the existence of God. His Theorem has become the basis of modern decision theory involving the calculation of probability of various hypotheses according to the existence of various weighted evidence. The interest lies in its use for modifying the probabilities of uncertain hypotheses according to the evidence. Bayesian inference is used in the PROSPECTOR system (Duda and Gaschnig 1978). In PROSPECTOR two ratios of likelihood are obtained from the expert

LS (measure of sufficiency)

LN (measure of necessity).

These are then used by the system to calculate posterior probabilities from the evidence provided. For example

IF x THEN (to degree LS, LN) y

The user can either state that;

x is definitely true

x is definitely not true

the user is uncertain whether x is true, but provides information on a scale of -5 (definitely not) through 0 (no preference) to +5 (definitely true).

In some systems (eg MYCIN) a variation is used which employs certainty factors of -1 to +1. In both cases the rules contain these measures which enable the systems to calculate for the consequences of the application of the rule. Such statistical approaches also allow cumulative certainty to be calculated as in the case of several interlinked rules, each with its own weighting.

Fox (1980) compared the performance of a Bayesian inference system with a heuristic non-probabilistic system and found that the latter matched human behaviour better than, and diagnosed as well as, the Bayesian system.

Certainty factors

Certainty factors allow the knowledge base author to attach a certainty factor, within a set range (-1 to +1, or in some examples +5 to -5) to the rule.

```
IF shirt_colour   = White
AND stadium       = White Hart Lane
THEN team         = Tottenham <0.90>
```

This rule leads to the conclusion that the football team in question is Tottenham with a certainty of 0.90. The certainty is not higher because visiting teams occasionally wear white shirts.

There can be little argument about the interpretation of a certainty factor of 0.99, but as Forsyth (1984) has pointed out, users can have different perceptions of 'unknown' and that a certainty factor of 0.6 could represent a number of levels of uncertainty about a truth. A further point of consideration is that the expression of any conclusion in terms of a numerical measure implies an air of precision which may not be justified. The extent to which MYCIN's rules and reasoning methods did not depend upon precise values of certainty factors is shown by Buchanan and Shortliffe (1984).

Other criticisms have been made of the approach of attaching numerical values to the degrees of uncertainty that they involve and then propagating these values through a sequence of deductions in order to arrive at a conclusion with a measure of likelihood or confidence factor. Hence the main criticisms of this approach include that;

a) it fails to distinguish between

different types of uncertainty (incompleteness or unreliability)

b) it is seldom clear how the numbers are derived and what their exact meanings are supposed to be

c) it makes it difficult to define conditions under which two pieces of information are inconsistent with each other

d) human experts are reluctant to attach numerical values to their uncertainties

e) numerical representations of certainty hide the reasons that produce them and thus limits the reasoning about uncertainty

There is an argument that 'certainty text' should be used instead of certainty factors or values. This means using such phrases as 'very likely' and 'very unlikely' instead using certainty values of 0.85 and 0.20. I do not see this as being a particularly significant argument. I accept that the novice user is probably more comfortable working with text than with figures, but I think that it is only a short distance along the learning curve before the novice user becomes accustomed to using a particular set of figures. The problem of discriminating between 'close' values still remains, whether it is phrased as text or numbers. However I do agree with the concept of Szolovits (1982), that it would be beneficial if this text could be generated directly from the data structures rather than being portions of canned text.

Further discussion about the issues involved in reasoning with uncertainty can be found in Szolovits (1982), Cohen and Grinberg (1983), Welbank (1983) and Ganascia and Kodratoff (1985).

TRADITIONAL METHODS

Conventional programming techniques have been used to create the large data processing systems which are more commonly associated with computers. Such systems are capable of collecting and processing vast quantities of data by means of complex algorithms. These algorithms are made up of step-by-step instructions that guarantee that given the correct data, the correct conclusion will be reached. Each time the processing takes place, the data may be different, but the processing follows the same predetermined route and results in the desired conclusion. Such a system is essential in, for example, a payroll program. Once they have started processing, conventional programs usually proceed on their own. On the other hand, the characteristics of expert systems, as discussed earlier, are highly interactive and tend to rely on heuristics rather than algorithms. They will accept and be able to use incomplete or uncertain information and can weigh up likelihoods, explore

alternatives and follow a course of reasoning which depends upon the user's replies rather than a preselected list.
Another major difference is the way that conventional programmers go about their task, in that they receive a system design (they know what the end product will look like) from a systems analyst, they produce a detailed design and then attempt to implement that design (one meeting between 'expert' and programmer may often be sufficient). In the case of expert systems, as a prototype is built, the knowledge engineer and domain expert meet more frequently as the expert is an active member of the development team. From the expert's comments on the prototype and further knowledge elicitation, the next version of the system will be built and refined and so on. Hence neither expert nor knowledge engineer knows what the final product will look like.

Waterman (1986) compared data processing with knowledge engineering, but it should be appreciated that knowledge engineering and data processing are complementary techniques which will find applications in most commercial organisations. It could be said that if data processing provides clerical power, then knowledge engineering provides intellectual power.

AI languages are not always needed, as expert systems could be developed in programming languages such as BASIC, COBOL, PASCAL etc. This statement is supported by the number of different languages that have been used to develop the systems analysed in Part Two. Programming itself is not an exact science, as there may be several means of producing the desired result and the choice of one method over another may often be a matter of judgement. Each programming language has its own particular strengths and weaknesses when applied to particular problems (horses for courses). As the strategy, heuristics and basic assumptions upon which traditional programs are based are not explicit in the program code, any mistakes made by them will be difficult to remedy, resulting in the higher cost of software maintenance. Some of the limitations of conventional languages are that the program could be more difficult to modify, they cannot handle uncertainty and contradictory evidence and they lack explanation facilities. These are problems which AI languages can handle. Expert systems do have the potential to learn from their errors and their problem solving abilities can be improved. In addition the language of a knowledge base is nearer natural language than many other programming languages.

Johnson (1984) observed that the use of AI languages allows programmers to work in a more generalised concepts by concentrating upon what has to be done rather than how it is to be done. However, there is not a clear border between AI languages and other conventional languages.

BUILDING AN EXPERT SYSTEM

Choosing a good application will make the difference between success and failure in developing your first expert system and probably the single most important issue, certainly as regards the development of commercial applications, is that there must be a pay-off. Hence the maxim to only use them when good conventional solutions do not exist and try to ensure that the expert system solution provides considerable value for the expense involved. Although there is almost universal agreement on the fact that there must be a pay-off, d'Agapayeff (1984b) pointed out that he believed that it would only be possible to produce a cost-benefit analysis of the expert system project for management to consider, after the project had been completed. Hence in establishing an expert system project, there is an element of a management act of faith. The criteria required for successful development, identified by Johnson (1984), included a favourable environment which obviously favours development by large wealthy companies with experience and resources and who can afford to pay, but also where the application will not make great changes to established practices.

Assuming there are sufficient tools and talent and the basic requirements for building a commercial expert system, as noted by Rees (1984) and Turner (1985), are available, it is also vital to remember that expert systems, as with all software systems, must be built on a firm base of software engineering principles.

Rees (1984) reporting upon his experience of DEC noted that management support was essential and it was also necessary to create effective management for the development and introduction process. One means of achieving the latter is to ensure end-user involvement from an early stage of development.

The technical difficulty involved in developing expert systems is a moving target because the technology is not static and so over a period of time, this factor may decrease. Indeed it is noticeable that the development of systems is no longer confined to the larger organisations.

Reporting on his attitude towards applications, based on his experiences at Unilever, Baker (1984) suggested that the problem is the key issue and that applications must be 'needs-led', a point echoed by Turner (1985). However, the success or failure of the project should not be crucial to the success of the organisation. The development of small systems as training exercises, even if they do not have significant practical value, can be a useful and valuable exercise. It was further noted that the use of systems must encourage expertise development and the methodology must be 'teachable'. The latter two points are particularly significant in that Baker is trying to encourage further exploration and development and to discourage the use of the technology as 'black boxes'. This thinking is soundly based as it ensures that an adequate base of expertise is created and that expertise is then an asset of the company which can be used to stimulate and underpin further development both within the company and as a form of consultancy. Baker (1984) predicted that

> "if we have the will, then knowledge engineering can be an everyday approach that is widely used throughout Europe within the next 10 years"

The fact that expert systems were being built to a relatively standard top-level architecture, with the knowledge represented in one form or another in a knowledge base which was separate from the inference mechanism, or control logic, led to the idea that the inference engine itself could be supplied with empty data structures, known as 'shells', which could be filled with whatever knowledge was required by the application. For example removing the rules for the diagnosis of infectious disease from MYCIN (Shortliffe 1976) yielded EMYCIN, variously described as (Essential Mycin or Empty Mycin or Engine Mycin). EMYCIN (Van Melle 1979) was subsequently used in the development of PUFF (Barr and Feigenbaum 1981) and SACON (Bennett and Engelmore 1979).

This simple 'shell' concept will only be successful provided that the shell's knowledge representation structure, which does vary from shell to shell, is suitable for the particular application. This has led to the production of a variety of commercially produced shells, each with its own particular strengths and weaknesses.

Basden (1983), Hayes-Roth, Waterman and Lenat (1983), Weiss and Kulikowski (1984), Hewett and Sasson (1986), Jones (1986a) and Waterman (1986) provide much relevant and accurate advice for the would-be system developer. Their recommendations have been structured in the form of a series of production rules. Notice that although it may be possible to build an expert system, it may not be justified or appropriate to do so. Indeed instead of asking whether an expert system will solve the particular problem, it would be more appropriate to ask what aspects of the problem lend themselves to expert system development.

<u>To build an Expert System ?</u>

IF system development is possible
AND system development is justified
AND system development is appropriate
THEN go ahead and good luck !

IF task doesn't require common sense
AND task requires only cognitive skills
AND task is not too large or difficult
AND task is not poorly understood
AND genuine experts exist
AND experts can articulate their methods
THEN system development is possible

IF task solution has a high pay-off
OR human expertise is scarce or being lost
OR human expertise is needed in hostile environments
OR human expertise is needed in many locations
THEN system development is justified

IF task requires heuristic solutions
AND task requires symbolic representation
AND task is not too easy
AND task has a practical value
AND task is of manageable size
THEN system development is appropriate

Note that the above list has not deterred many from attempting to develop systems where, for instance, experts do not agree, but it does provide a sound basis for the initial decision of whether to attempt to build an expert system. However, the matter may be far from the above representation, as Forsyth (1988), quoting two 'rules' of knowledge engineering, provides further advice.

> "Rule: 'Scylla'
> IF the knowledge is easy to formalise
> THEN the application is trivial
>
> "Rule: 'Charybdis'
> IF the application is interesting
> THEN the knowledge is hard to formalise"

Having decided that it is possible, justified and appropriate and having ascertained the most efficient approach then decisions about the most cost-effective computer resources must be reached. Already some of the decisions that have been reached will have narrowed the decisions on, say, what hardware to use. At this stage there will be a need to trade off hardware and software costs and building strategy. As the technology improves and the price falls, this latter consideration may become less important.

The initial design

Start off with a small project, but one that will have a pay-off, as it is easier to start with a small project with limited aims rather than embarking on a more complex project. The latter would appear to have a greater risk of failure, often due to subjective human reactions rather than any fault in the software. Nevertheless, a big expert system is not a little expert system that has grown up and a small expert system is not a big one that has been butchered. The problem of the incremental nature of the development of the knowledge base has been called the paradigm shift (Hayes-Roth 1983). This is the point where the size of the knowledge base becomes too large for effective use and at this point, it will be necessary to redesign the system. Hence, both large and small systems need an appropriate level of planning from the outset.

The advice provided by Hewett, Timms and d'Aumale (1986) for developers of expert systems was to go for a system that would involve between 50 and 100 rules. This is sound advice, except that it is difficult to estimate in advance, the number of rules in the final knowledge base. Also rules can be refined and what may have started as four or five rules may be able to be finally expressed as a single rule. Conversely, it is vital to ensure that in the refining process some small detail which may, under certain circumstances, however rare, be an important part of the system, is not eliminated. For a successful application, it is suggested that the security of a low-risk development environment is maintained and a conservative approach in predicting the capabilities of the system is adopted. Note that it is easy to overestimate the capabilities of a proposed system and end up with a disappointed audience and little support for future projects. (Remember Lighthill (1973) as quoted in chapter six).

Finally it would be sensible to choose a domain that has relatively easy access to the domain knowledge and which will be easy to understand, as a certain amount of learning about the domain will be necessary.

Tool selection

The choice of an appropriate tool will be a matter of compromise between resource requirements and the required flexibility of implementation. As it is difficult to predict exactly which software features will be needed, choosing one with a variety of features, but which is also easy to learn, would be a sound initial maxim. Hayes-Roth et al (1983) suggest that the problem characteristics determine the tool selected and that the tool should be tested and evaluated early on by building a very small prototype system.

The choice of tools lies between languages which offer greater flexibility, but which require longer development time and greater expertise on the part of the system builder and shells which have a rigid structure and as a result are able to offer much faster implementation times, but lose out in terms of flexibility. The use of toolkits, which offer a compromise between these two extremes, is another possibility. A further compromise that could be made is to use a shell to provide a rapid prototype for evaluation purposes and then build the final system using a programming language.

Commercial developers will make decisions about tool selection for a number of reasons. Johnston (1985a) and Becker (1985) provide two case studies of large developments with the former (Plessey) selecting languages and the latter (ICL) using AI workstations.

In principal any language can be used to write an expert system. The relative advantages and disadvantages of using AI languages were summarised by Hewett, Timms and d'Aumale (1986). However, as with conventional programming, some languages lend themselves more easily to particular types of applications. In particular LISP and PROLOG have architectures which are more suited to expert system development. PROLOG (PROgramming in LOGic) was developed at the University of Marseilles in the early 1970s. It is a declarative language in that it states what is to be done rather than how it is to be done, as is the case in traditional programming. Further details can be found in Clocksin and Mellish (1981), Clark et al (1982), Ennals (1983), Clark and McCabe (1984), Conlon (1985), Brough et al (1985) and Bratko (1986).

LISP (LISt Processing) was developed in America in the 1960s and a LISP program consists of a list of functions (in the form of lists) which are applied to arguments. Winston and Horn (1981) and Norman and Cattell (1983) provide further details.

Although the theory of the use of a shell is that you buy a shell and construct a knowledge base, the reality is not quite as simple as that. The structure of the knowledge base and the inference mechanism will have been designed for a particular use and it may not be universally acceptable. Nevertheless there are a variety of shells on the market and selecting an appropriate one is an important consideration.

An extension of the facilities of the programming languages, but without going as far as the shell is provided by toolkits and operating environments, based on 'AI workstations' which are powerful and expensive super-microcomputers. Among the features provided by these machines are powerful and fast processing speed, windowing and sophisticated graphics.

Waterman (1986) noted that for every tool there is a task perfectly suited to it. However, the converse is not necessarily true, there may be a number of tools that would perform to an equal level. It may also be the situation that none of the tools is perfectly suited to the task.

The software directory in the NCC Expert Systems Resource Pack (NCC 1987) contains full details of current expert system building tools. However, Waterman (1986) provided the reminder that expert system building tools are not good at performing the knowledge acquisition task, refining their knowledge bases and handling mixed representation schemes.

Prototype production

To maintain interest in the project and perhaps convert some of the doubters, it is important to build a demonstrator system which should be up and running in a relatively short time. The effort required to produce a prototype was estimated by Hewett, Timms and d'Aumale (1986) as being about 10% of that which is needed to develop a large system. A good domain for a demonstrator is one which is relevant, but not critical, to the central activities of the organisation and it should be seen as beneficial and a positive step, not just a fancy research idea. An area which has not, or could not have, already been addressed by the organisation may highlight the potential for developing an expert system.

Finally it is essential that there is the enthusiastic commitment of a human expert, because without this, or a lukewarm version, the project will be doomed to mediocrity or failure. This is one of the few things about which there is almost universal agreement amongst the expert system community.

The argument that you need special AI workstations is no longer valid as PC-based shells make it feasible to quickly have a working prototype which can then be corrected, enhanced and developed. This is a highly iterative process. The prototype will also demonstrate the limitations that the system will never be 'perfect' and it is important to define a 'cut-off' point where the system will be working at an agreed level of effectiveness and any further development work will not result in any cost-effective real improvement. The 80:20 rule states that 80% of the functionality of a system can be produced in 20% of the time. It is the remaining 20% of the functionality which takes the time, but it will probably be that final 20% functionality that will justify the development of the system. Finally, it must be remembered that applications are not guaranteed to be successful.

Implementation

Buchanan and Shortliffe (1984), based on their experience with MYCIN, made several recommendations regarding the implementation of systems. Striving to minimise changes to current practice and considering the concerns and demands of end-users were seen as important considerations. There is the potential for organisational change that will be created by the introduction of expert systems, but any systems which avoid requiring or creating any organisational change when first introduced will stand a better chance of implementation. The new technology will be enough to cope with and change at any time is threatening and has wrecked many a project. However, this is an aspect that should be borne in mind by the developer, perhaps initially keeping it in low key.

Two further considerations are concentrating on enhancing the interactive capabilities of the system and recognising that 100% accuracy is neither achievable nor should be expected. Turner (1985) reporting on the lessons to be learned from his experiences in developing ESCORT, noted that the development of a system like ESCORT is never complete.

Hawkins (1983) pointed out that if any system is going to be able to satisfy the majority of the expectations from a human expert, the system builders must consider some of the following issues. Human experts tend to adjust their dialogue to their users, the expert system has to be able to understand the significance of every interaction and produce an answer, and in a language, that can be understood by the user. Any advice or results must be justified and explained on request, but this is not necessarily the same as merely producing a rule-trace. The system must also be able to recognise the fact that conflicts do exist, recognise it when one arises and provide appropriate advice on the handling of the conflict.

Careful consideration should be given to choosing the most appropriate criteria for assessing the system. Testing expert systems may be a particular problem, as identified by Jones (1986), especially where there is induced knowledge which itself cannot be easily verified. Additionally, as noted by Stock (1988)

> "an expert system is a human-machine system, its success cannot be measured solely by the performance of one half of the system."

The basis on which an expert system should be tested or validated, relates, therefore, not just to the knowledge base contained within the system, but also to how it is used by both the program and the end user.

PART TWO

COMMERCIAL

APPLICATIONS

CLASSIFICATION OF EXPERT SYSTEMS

Any classification task can be approached from a number of directions and any attempt to prescribe techniques to problem characteristics presupposes that such a general scheme of classification exists. Even the following examples do not represent a mutually exclusive taxonomy for describing expert systems. In most cases they will defy neat categorisation, the expert systems may have one major characteristic but will also certainly have aspects of many others. This is certainly something that I have found as I attempted to classify the entries in the systems that are analysed later in this section.

The approach taken by Hayes-Roth (1983), Chandrasekaran (1984) and Waterman (1986) is to classify systems on the basis of the task undertaken. Rychener (1985) presents an alternative classification scheme involving three types of problem (diagnosis, design and planning) on a

'Given-Goals-Constraints-Operations' basis. Providing a classification according to the search strategies employed by the system, in a much quoted work, Stefik et al (1982) view expert systems as problem solving programs and identify large solution spaces, tentative reasoning and noisy and time-varying data as issues which appear across the catalogue of expert system tasks.

Hewett and Sasson (1986) choose a simple scheme based on two characteristics relating to the nature of the task. The first of the two characteristics is concerned with whether the task involves classification or creation which distinguishes between those applications which are generically termed classification from those termed design or planning applications, which have a more creative nature.

The second characteristic concerns whether the data remains fixed for the duration of the system's operation. In, for example a configuration task, the data is fixed and the list of components will not change, at least for the duration of the task. In a real-time control application, the data will be changing as circumstances change and as a consequence, the nature of the task may have been changed. In the former example the conclusion reached by the system should be constant, whereas in the latter case, the conclusions will be changing in the light of data changes. Hence real time design and planning will be the most complex systems.

The data may be static during the system's run so that parameters, that will not change, can be set at the beginning (a small solution space). Dynamic problems may be far more complex depending upon whether the changes can be predicted. If the data is changing, but within specific predictable limits, then this can be modelled within a larger, but manageable solution space. If the data changes are unpredictable then the solution space rapidly becomes vast and potentially unmanageable.

This produces a four-state matrix of categories;

static classification	real-time classification
static planning	real-time planning

The main difference between the four states in the Hewett and Sasson (1986) model concerns the solution space. All systems in the former group have a limited solution space that is known to the system and the problem is one of classification (eg fault diagnosis, cataloguing, prescription, training and counselling or instruction). In contrast, design or planning systems have an unknown solution space and need to be forward chaining, as the system evaluates various possibilities using rules to select the 'best' solution (eg configuration and planning systems). The technology frontier is moving across their classification as, despite the complexity, an increasing number of real time systems are developed.

In this thesis I am attempting to use a dual approach and classify the systems according to the domain in which they work and according to the type of task that they perform. This follows Reichgelt and van Harmelen (1985) who use both task-related and domain-related criteria. A further reason for my choice is to examine the concept of 'islands' of expert system development (Johnson 1984). Johnson's argument was that early investment would be restricted to a number of application 'islands' where conditions for such development would be favourable. These favourable environments were to be found in wealthy industries who could afford to invest in the technology, even if there was not a significant commercial pay-off. Computing, electronics and communications, financial services, military applications, oil exploration and extraction were specified as being promising 'islands'. As was discussed in chapter thirteen, the suggested reason for the majority of commercial investments in expert system technology was in search of a competitive advantage. The initial cost of research and development would be likely to limit the number of pioneering organisations as involvement with the technology was not accompanied by any guarantee of success.

Johnson's thesis seemed to be correct as a number of major companies (e.g. Unilever, Shell and ICI) built up expert system teams to investigate and develop the technology. His model does not now hold up so well, as the areas of development have advanced in all directions. Cost appears to be the main factor here and it is a measure of the speed of development in this area that since 1984 the costs of both hardware and software have dropped. The demystification of the technology and increased ease of access to the technology has enabled the expert system market to expand. A more up to date assessment of the present position, as described by Hewett, Timms and d'Aumale (1986), is such that 'healthy' companies are actively looking to the technology to provide solutions to their problems, rather than merely spectating upon developments. The idea that they had to be big and complex has waned as many comparatively small systems have been developed and demonstrated as being useful to the developing organisation. d'Agapeyeff (1984, 1987) frequently propounds the benefits to be obtained from small applications.

As the frontier of technology has shifted, so has the focus of developmental difficulty. In the early eighties, the major problems were largely technical, but now the main restraint on building systems is likely to be a lack of experience within the company. If the company can afford the investment required to set up an in-house development team, this may be an option. However if this is not possible, then it is reasonable to buy a ready-made system or an easy to use shell from one of a number of the software companies that have become established to cater for this market (eg Expertech, Intelligent Environments, Expert Systems International).

What tasks can expert systems undertake ?

Skuce and Matwin (1985) classified expert systems into three groups, the recognisers, the designers and the advisors. Recognisers are systems that classify a situation into a known category given appropriate evidence, designers follow a specification and produce a design configuration in some task. Advisory systems, which are discussed in more detail in chapter eighteen, give information for decision making.

It is relatively easy to describe the following tasks, based on Stefik et al (1982), Hayes-Roth et al (1983) and Goodall (1985). However it is difficult to provide clean and clear-cut divisions between the various tasks that expert systems may be asked to perform. For example diagnosis systems may often be used in conjunction with debugging and repair systems and both monitoring and diagnostic systems attempt to detect faults or malfunctions, but diagnostic systems lack the iterative aspect. Similarly, control systems may well include components designed to perform many of the list of tasks (eg monitoring, prediction and troubleshooting). Troubleshooting itself is a sequential combination of diagnosis, debugging and repair. Brown et al (1982) noted that SOPHIE, a system whose main task is instruction, uses simulation in order to perform prediction and 'what-if' analysis.

Advisory systems provide advice to assist in the decision making process. This category of applications is further developed in chapter eighteen.

Control systems are treated separately as they provide a small and limited task specification. However, they must include a monitoring component and are likely to include components to perform other tasks. The systems work by governing the behaviour of a system through modifying specified parameters to maintain the settings of various devices within prescribed limits.

Classification tasks include debugging systems for producing remedies for malfunctions. They often incorporate a diagnostic feature to isolate the cause of the specific malfunction which may be the root cause of several seemingly unassociated malfunctions. The main problem is that the remedy may have associated constraints. Instruction systems contain a diagnosis, debugging and repair facility for students which model what the student actually knows and matches that to the 'ideal' model built into the system, deficiencies and malfunctions may then be remedied. Diagnosis problems are typified by the need to search through a fixed set of possibilities either for the one 'correct' result or to advise on viable alternatives. Often the rules regarding the 'choice' are qualitative in nature and cause-effect relationships may well be 'fuzzy'. Faults, which may be intermittent, may be masked by symptoms or by other faults. Repair systems provide a remedy for a diagnosed malfunction, however, few operational repair systems have been developed because of the complex nature of actually executing 'debugging' plans on real objects (eg SPECT).

Interpretation systems provide the analysis of data to determine their meaning or inferring descriptions of situations based on information received from sensors. These systems often work in real-time with sparse and unreliable real data as opposed to the 'clean' symbolic representation used in many other tasks. The key problem in this area is seen as 'noisy' data (missing, erroneous or extraneous). In monitoring tasks, the system compares, either continually or periodically, actual to expected observations and to set off an alarm when intervention is required. Under such circumstances, what constitutes an alarm condition is often context dependent. Jones and Davies (1986) observe that, by definition, these systems must deal with time (often real time) and make their decisions based not only on context, but also on time.

System complexity is the key limitation for design systems which arrange and organise (configuring) objects given various constraints. Fox (1986) has shown that design tasks can be broken down into at least four categories;

a) selection, mapping requirements to attributes

b) configuration, where the number of options in the problem space is too large to solve by selection

c) extrapolation

d) discovery

Such systems often synthesise partial designs and simulate or test these against the original design specification. The sequencing of assembly instructions means that there is also likely to be a planning element. As many problems require reasoning about spatial relationships, this provides further task constraints. Additionally, the designer may not be able to assess immediately the consequences of design decisions.

The creation of a program of actions that can be carried out to achieve goals is the aim of planning systems. The requirement is for the application to be able to model the system itself and the inter-relationship of the component parts, to evaluate alternative courses of action taking due account of strategies and also deal with exceptional conditions. The system is given a goal with a set of any constraints and it has to produce the 'best' possible feasible solution. As they need to produce a complete course of action, these systems must be able to backtrack in the light of problem constraints. Backtracking can be costly in terms of time and/or memory and some planning systems break the original problem into sub-problems to avoid having to replan everything when it comes up against a dead-end. Tate (1985) comprehensively describes planning tasks and identifies the following as key issues;

a) the problems are frequently large and complicated

b) the planner may not understand all of the consequences of his actions, hence planners must act tentatively

c) the planner must be able to focus on the most important consideration especially if the detail is overwhelming

d) the planner must attend to interactions between plans for different sub-goals which are a feature of large complex problems

e) the planner must plan for uncertainty as the planning context will only be known approximately

Scheduling can be viewed as a type of planning task that involves synchronisation of resources and hence a significant time element.

Simulation is a method commonly used by Operational Research staff to model a complex system involving changes over time and prediction may also play a role here. The aim of prediction systems is to infer the likely outcome of given situations, often having to make use of diverse data, and to forecast the course of the future from a model of the past and present. Stefik (1978) noted that this is a particularly complex task because

> "it requires reasoning to allow for multiple possible futures with undetermined operations, unordered sets of possible future events and the possible actions of uncontrolled multiple actors."

A further task of particular interest to education and one which I believe will become increasingly important is knowledge retrieval, where knowledge is encoded for future use as in rare skills archiving or text animation (eg ES/P Advisor).

THE POTENTIAL OF EXPERT SYSTEMS

This chapter introduces the basic characteristics of the market place for expert systems. There is a fairly typical cycle of commercial and non-commercial activities which can be applied to the development of expert systems. The sequence of actions in the developmental cycle can be represented as follows;

no action
watching brief
investigation
demonstration systems
developing systems
larger projects
integrated systems

or alternatively

awareness
interest
evaluation
trial
adoption (or rejection)

Hewett, Timms and d'Aumale (1986) showed that the development of leading organisations in this field (eg computing, electronics, defence and aerospace) took six years (1981-87) from initial interest to the first large scale operational systems. Other advanced organisations (eg large financial and manufacturing) were working on a 1984-89 development timescale with the mainstream organisations from 1986-91.

If this pattern of development is accurate, then it can be justifiably claimed that the technology has survived its infancy and represents an example of technology transfer from academic work to commercial application. The analysis of actual usage in the following chapter would appear to support this premise.

Expert systems are useful in problem solving where the information is largely in the form of heuristics, the qualitative analysis of the problem is more important than the quantitative analysis and the route to the solution is as important as the solution itself. The characteristics of expert systems suggest that they could be used in the following generalised situations;

a) where knowledge is expressed with certainty

b) where much information is provided and this is not necessarily a fixed quantity and a great deal of knowledge must be consulted

c) where every possibility must be explored

d) where databases of facts must be referred to

e) where preliminary consultation would help to prepare for a meeting with a human expert

Having ascertained that one or more of those conditions exist, the company should look at further strategic and economic considerations. In situations such as these, it is the job of the knowledge engineer to identify the situation and provide the optimum system tailored to the organisation's requirements.

But why use expert systems ?

Commercial companies applying AI technology to their applications are doing so to gain a competitive edge by improving their product, by improving their efficiency or by increasing their product range. This is a comparatively short-term commercial aim, whereas theoreticians have a much more long-term set of aims. Both types of approach are required to push the frontiers of this technology ever outwards.

Hewett and Sasson (1986) identified four key points as good indicators of success;

a) the task needs to be well understood

b) the task should take a human between half an hour and half a day

c) the knowledge for performing the task should be based on heuristics

d) there must be a pay-off. However it may be difficult to prove or quantify the pay-off, particularly in control applications where it may only be when something goes wrong with the human expert, that the advantages, or disadvantages, of having the system are seen.

The first three are necessary task

characteristics, but without a well defined return on investment, a system cannot be considered as being commercially successful. This return on investment may be identified as increasing expert productivity or capability, tackling projects which could not be developed with conventional techniques (eg XCON, SHUT1) or perhaps preserving expertise within the company eg SOUP.

Johnson (1984), Goodall (1985) and Myers (1986) provide the basis of the following summary of reasons for their use.

<u>They can increase profitability</u>

In purely commercial terms, this is undoubtedly the prime concern and this may mean saving time and/or money (eg DRILLING ADVISOR) or working with cheaper equipment (eg DENDRAL). Increased profitability may arise as a result of being able to speed up certain processes. Becker (1985a) reported that NAVEX is planned to complete within a couple of hours, the task that presently takes between four days and four weeks. Many small professional firms, particularly in law and accountancy, will already employ a team of staff with expertise in their own specialist fields, but will be unable to take on work in other areas where they lack the necessary expertise. Expert systems provide an opportunity for such firms to expand their business without expanding the existing staff. The systems may add flexibility, which would be particularly important where there may be rapid change in the industry (eg XCON).

Although there may be the opportunity to do a job with fewer staff, this may be a double-edged sword with a negative side in that the 'surplus' staff are merely 'removed' or more positively, in that they remain in employment as a result of improved business as a result of the system perhaps increasing turnover. Chapter thirty two discusses this social dilemma in more depth.

They can provide an extension of human capabilities

Rauch-Hindin (1986) reported that the use of expert systems can enable experts to work at a level that uses more of their capabilities, suggesting that those who work at 75% of their capabilities could be raised to 85%, those at 85% to 90% and those at 90% to 92%. These figures may be small in percentage terms, but, if realisable, they may be commercially significant in terms of time or money. There is a further point in that expert systems provide the opportunity for 'low level' staff to operate at 'high level', which may produce a greater reliance upon such junior staff. In the long term this could prove to be detrimental, an issue which is discussed in chapter thirty two.

Such systems may assist towards qualitative improvements in human performance as to methods, procedures or judgements (eg prompting the consistent use of appropriate methods) or in quantitative terms. Compilers

and interpreters, without which all computers would be totally useless, could be considered as expert systems. The compiler must be able to translate the high level language into machine useable code. It must therefore have the expertise to recognise not only the valid, but also the invalid, syntax of the language, check it for consistency and then produce the resulting machine code.
Programming is essentially the application of rules and regulations to translate the requirements of the user into terms that the computer can 'understand'. Glassup (1985) expects the first pay-off of this technology to be to make writing systems cheap, which suggests that it

> "effectively bids farewell to the programmers and analysts"

This may sound a little extreme, but time alone will tell if this is going to prove true. Quoting a figure of a 150% increase, Rauch-Hindin (1986) suggested that expert systems can allow an increase in programming productivity, but Johnson (1984) notes that this is one area where such claims need to be proved in routine use. Ince (1988) reported that in some applications, instead of replacing human consultants, the expert system is replacing program code and these rules can be easily modified by non-technical staff.

Professional expertise is expensive to obtain in terms of both time and money. Simon (1983) reported that it takes a human twenty six years to acquire sufficient knowledge to become a computer scientist. If the expertise can be 'taught' to a computer program, then the resulting software can be rapidly duplicated at minimal cost and the benefits felt by millions of users. The medical problems of the Third World seem to provide one example of where this approach would be of great benefit.

Expert systems have the potential to be as effective as the better consultants and more effective than most. There is a spectrum of possible uses ranging from making additional skills available to professionals and thereby improving the accuracy and efficiency of existing experts, to teaching expert skills to aspiring professionals and making everyman his own expert.

Where the system is used in place of a specialist, Basden (1983) and Duke (1985) identify reliability, accessibility and consistency as the important benefits. Humans can forget relevant factors especially in areas in which stress or urgency is involved, whereas systems can handle large volumes of data and will not overlook a situation (eg SUS).

The human expert may not be able to work in particular places, whereas expert systems can function in hostile environments. Alternatively, the human expert could be anywhere so increased accessibility and the easier duplication and wider distribution of expertise could lead to the dissemination of real experience as against academic theory.

Improved consistency would be demonstrated in situations where there is uncertainty in observation, paucity of data, or questions of probability or the relative importance given to different factors are primary considerations. An expert system could act as an aide memoire, the advantage of using an expert system rather than a printed checklist is that the question order of the latter is fixed whereas the expert system could 'intelligently' select the order of questions depending upon circumstances.

As with all computer applications, there is the further advantage of the ability to arrive at a faster solution or try a greater number of possibilities in the given time. Systems can perform better than a human because they make fewer mistakes and do not become tired or bored (eg PROSPECTOR). However, Rauch-Hindin (1986) reported that expert systems do not possess the 'gut feeling' that many experts develop as a result of years of experience and use in their everyday work.

Finally there is the 'immortality factor', what if we had been able to capture in an expert system the powers of Einstein, Newton or da Vinci ?

<u>They are tools for manipulating knowledge</u>

Many bureaucratic tasks are already well governed by rules and regulations and are very suitable applications for expert systems. For example systems to manage complex documents (eg DHSSD) or to explain a manual or reference document (eg FEASA). Systems such as REVEAL, can help to analyse knowledge and constantly self-improving systems could highlight weaknesses in current understanding.

In addition to some expert systems being able to handle uncertain knowledge, systems can codify and make explicit old knowledge (eg TEIRESIAS), can preserve expertise (eg SOUP, COMPASS) and discover new knowledge (eg AM). The ability to capture rare or disappearing expertise (rare skills archiving) is particularly significant where there is a high staff turnover either by design (as in the Forces) or as a result of other factors.

When acting to transfer expertise, the system could act as a medium for communicating expertise or know-how between experts of similar level or as a medium for pooling expertise from several experts in order to generate better or more consistent conclusions than may be reached by a single expert. Alternatively, perhaps to provide a tutorial in some domain, the system could act as a consultant, providing advice to the end-user in a similar fashion to a human expert. In such training applications, the 'WHY' facility would be of particular significance.

Although high performance is a commonly quoted characteristic of expert systems, Sridharan (1978) observed that

> "the ultimate importance of an application may be less in the production of a high performance program, but more in formalising, structuring and making known the private knowledge of a group of experts."

The applications of expert systems in the 1980s have verified the thoughts made during the 1960s by perceptive individuals such as Feigenbaum or Simon, whose main message was one of encouragement in that AI was possible and would have a major impact.

Availability, consistency and comprehensiveness are some of the advantages of expert systems. However there remains a longer list of problems still to be solved including knowledge acquisition and updating, technical limitations, testing and behaviour of systems, the choice of domain and, not least, human acceptability.

AN ANALYSIS OF THE COMMERCIAL USE OF EXPERT SYSTEMS

Expert systems are now appearing in more and more widespread use and it is the intention of this chapter to look at the many and varied uses and some of the problems and advantages of applying the technology. Throughout this chapter, I will be referring to my own analysis which is based on a database of 785 expert systems which were 'collected' over an eighteen month period starting in January 1986. Hence, in looking at the figures, it is important to bear in mind that 1987 does not cover a full year. Although it is a large database of applications, the figures may be biased towards UK and USA as the entries were largely taken from reports in the commercial and academic press.

The growth of applications

Banks (1986) reported on 'Winston's Curve', a step-like graph which plots applications

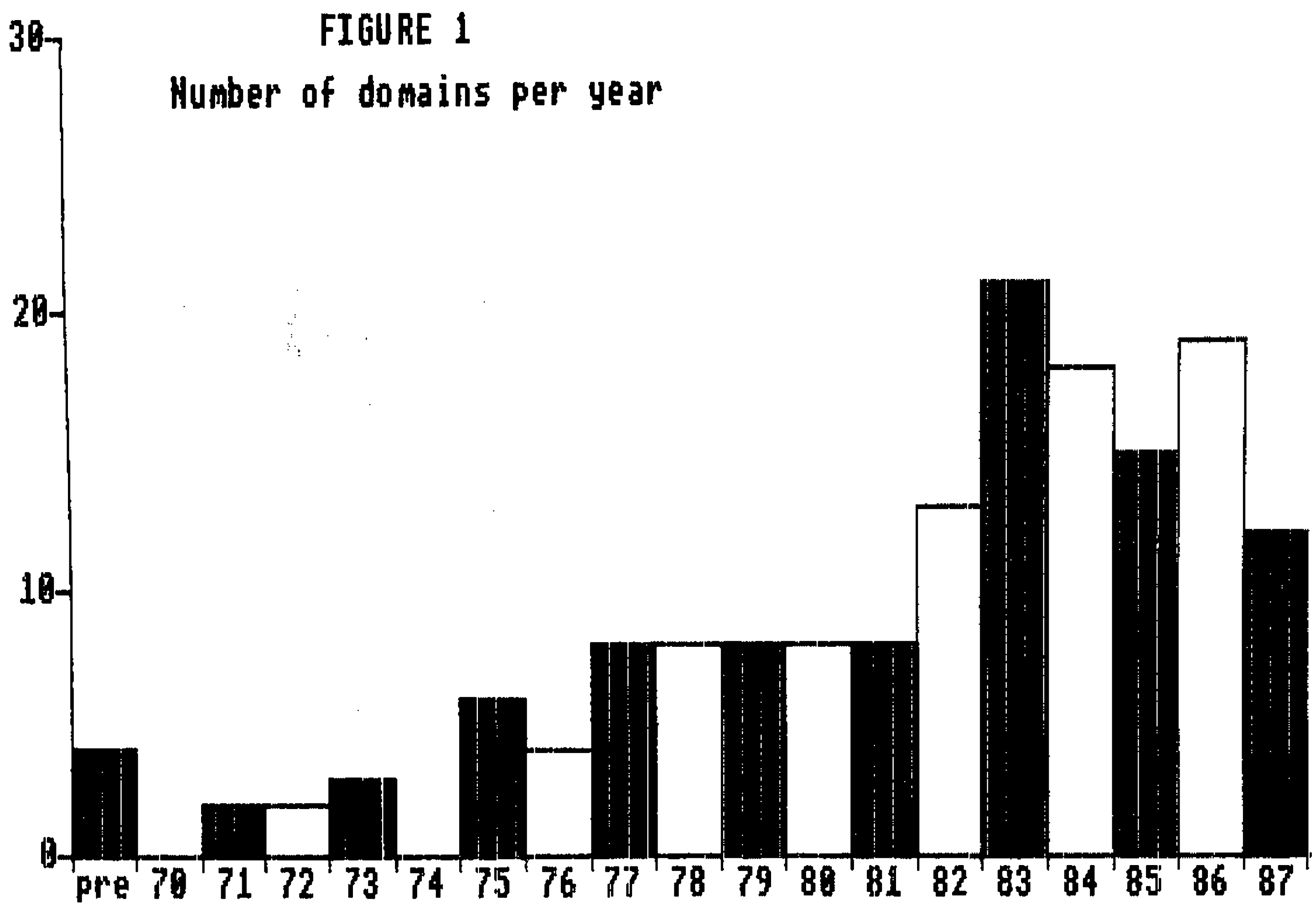
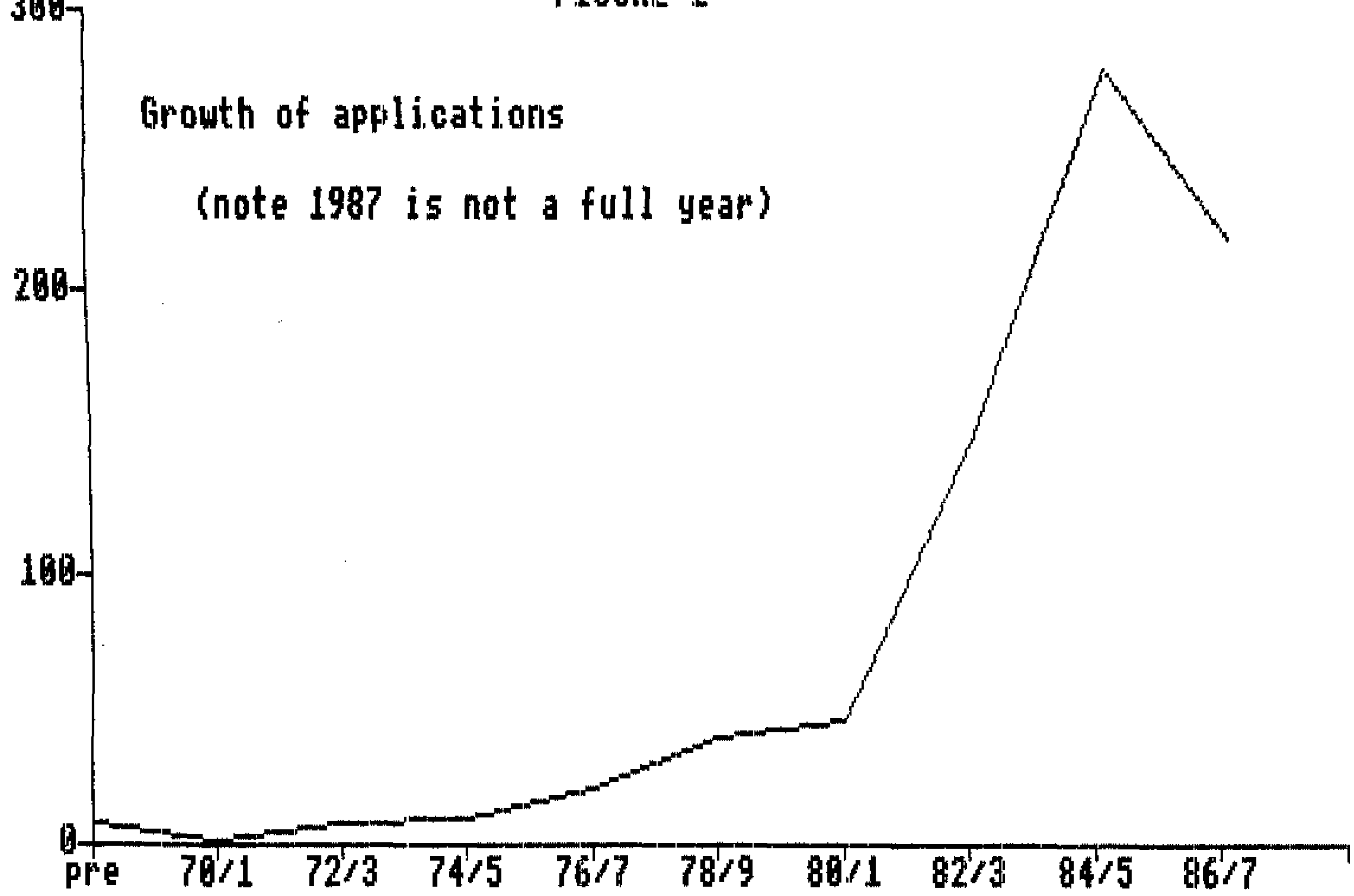


FIGURE 1
Number of domains per year

against acquired knowledge. Up to the end of 1985, the curve has been moving along a flat path along the knowledge axis. From the beginning of 1986 there is a strong step function up the application axis. Winston observes the net result of this will be that the academic community will soon run out of existing knowledge and the number of different applications will slow down. Time will tell whether this observation is correct, but figure 1, from my own analysis, provides supporting evidence. It indicates that after a slow increase in the number of domains where expert systems have been developed during the early 1970s, there was a 'step up' towards the end of that decade. A further increase is apparent in the early 1980s, but since then the number of domains has remained relatively stable. This is in stark contrast to figure 2 which plots the number of applications over time and clearly shows a steady increase in the number of applications up to 1980, followed by a sharp and dramatic rise.

FIGURE 2


Growth of applications
(note 1987 is not a full year)
300
200
100
0
pre
70/1
72/3
74/5
76/7
78/9
80/1
82/3
84/5
86/7

Further evidence of the spread of applications is provided by the calculation of a task/domain factor. Based on my figures, if the number of applications is divided by the number of domains the following 'task/domain' factors are obtained;

time		factor
to the end of the 1970s	=	2
late 1970s - early 1980s	=	3
1983 and 1984	=	5
1985 and 1986	=	10

A grouping of systems according to the Hewett and Sasson (1986) four state model

	static	real-time
Classification	219	74
Design/ Planning	244	29

supports their 'moving technology frontier' description and also their premise that real-time expert systems are more complex and will take longer to implement. However, implementation is not just concerned with technical problems. Sacerdoti (1982) pointed out that the environment into which the application is to be placed must be taken into account and identified several barriers of a sociological and economic, rather than a technological, nature that stand between the laboratory prototype and the commercial product. For example, a potential product must compete against alternative approaches and there is also people's natural resistance to change which is further discussed in chapter thirty two.

Task specificity

The analysis shows a high degree of task specificity in the applications of expert systems. On a matrix of 15 generic tasks against 25 task domains (375 possible entries), 67% of the entries are blank and 20% contain 3 or less applications. Therefore the applications are very largely confined to 47 entries which identifies where there is a close match between domain and task. For example 55% of all medical applications are diagnostic tasks and 49% of all software computing applications are programming tasks.

ANALYSIS OF SYSTEMS - TASK v DOMAIN

	Advisor	Control	Diagn	Class	Tutor	Debug	Repair	Design	Prog	Plann	Simul	Synth	Pred	Monit	Interp	Totals	%
LAW	16	0	0	0	0	0	0	0	2	0	0	0	0	0	2	20	3%
FINANCE	35	0	2	0	0	0	0	0	3	0	0	0	3	0	3	46	6%
BUSINESS	24	0	0	1	2	0	0	0	0	5	0	0	3	0	0	35	4%
MANUFACTURING	6	6	6	0	0	0	0	4	0	13	1	0	0	0	1	37	5%
IMAGE PROCESSING	1	0	0	0	0	0	0	0	0	0	0	0	0	0	4	5	1%
COMPUTING software	30	1	3	0	9	5	0	1	51	2	0	0	1	0	2	105	13%
COMPUTING hardware	6	1	24	0	0	0	0	27	1	9	0	0	1	2	2	73	9%
ENGINEERING	26	10	48	0	2	0	3	26	0	10	0	1	3	11	9	149	19%
AEROSPACE	1	4	4	0	1	0	0	0	1	5	0	0	0	5	1	22	3%
MILITARY	11	3	1	0	0	0	0	0	0	5	2	0	1	2	4	29	4%
PHYSICS	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	3	0%
CHEMISTRY	2	0	0	0	0	0	0	0	0	2	0	0	6	0	4	14	2%
METEOROLOGY	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	2	0%
OCEANOGRAPHY	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	0%
SCIENCE	1	0	0	1	0	0	0	3	0	0	0	0	0	0	0	5	1%
GEOLOGY	5	0	1	0	0	0	0	0	0	2	0	0	2	0	5	15	2%
MATHS	2	0	0	0	2	0	0	0	0	0	0	0	2	0	0	6	1%
AGRICULTURE	10	1	2	0	0	0	0	0	0	1	0	0	0	0	0	14	2%
BIOLOGY	1	0	0	0	0	0	0	0	0	0	0	0	3	0	0	4	1%
MEDICINE	30	1	82	4	8	0	0	1	3	6	0	0	1	3	10	149	19%
TRAVEL	3	2	0	0	1	0	0	0	1	8	0	0	0	0	0	15	2%
ARCHITECTURE	3	0	0	0	0	0	0	9	0	3	0	0	1	0	0	16	2%
ARCHAEOLOGY	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0%
EDUCATION	0	0	1	0	5	0	0	0	0	1	0	0	0	0	0	7	1%
VARIOUS	5	0	0	0	0	1	0	1	0	2	0	0	1	0	1	11	1%
Totals	219	29	174	6	30	6	3	72	62	74	3	2	31	23	51	785	
%	28%	4%	22%	1%	4%	1%	0%	9%	8%	9%	0%	0%	4%	3%	6%		100%
	Advisor	Control	Diagn	Class	Tutor	Debug	Repair	Design	Prog	Plann	Simul	Synth	Pred	Monit	Interp	Totals	

The development of advisory systems can be clearly seen in the business, finance and law sectors. Although on a smaller scale in terms of actual systems, very comparable in percentage terms are the agricultural advisors (71%), the travel planners (53%) and the architectural design systems (56%).

Such a matrix configuration of domains against tasks, both arranged in, say, alphabetical order, will produce 'islands'. A new and different set of 'islands' will be produced if the arrangement is randomised. However this approach does not help in the analysis of the database. Therefore, following the discussion concerning the classification of tasks in chapter fourteen, I have grouped together tasks. Similarly, I have grouped together those application domains where there are definite links. I selected the following five task category headings by combining together those tasks which had similarities. (The figures indicate the percentages of the total systems)

A) Advisors (28%)

B) Control (4%)

C) Classification (28%)

Classification	Debugging
Diagnosis	Repair
Tutor	

D) Design (30%)

Design	Prediction
Programming	Planning
Simulation	Synthesis

E) Interpretation (9%)

Interpretation
Monitoring

The domains were grouped according to the following table;

1) Business (13%)

Business	Law
Finance	

2) Engineering (54%)

Computing	Manufacturing
Aerospace	Military
Engineering	Image processing

3) Science (28%)

Medicine	Biology
Physics	Chemistry
Meteorology	Geology
Oceanography	Maths
Agriculture	

4) Miscellaneous (5%)

Architecture	Travel
Archaeology	Education

A matrix of these four domain groups against these five task groupings, highlights five 'islands' (1A, 2A, 2C, 2D and 3C) which make up 67% of the total applications.

	A	B	C	D	E	total
1	75	0	5	16	5	101
2	81	25	106	165	43	420
3	52	2	100	36	24	214
4	11	2	8	27	2	50
	219	29	219	244	74	785

These major 'islands' also hide some more localised 'islands'. Within 2C, for example, 72 of the applications are diagnostic in the domains of engineering and computer hardware and within 3C, 82 are medical diagnosis applications. In the largest island (2D), 53 are design applications in the domains of engineering and computer hardware, 51 are computer software programming applications and 49 are planning tasks.

Taking 1984 as a watershed, from which to further analyse the development of 'islands', shows the fall in 'science' applications (Group 3) in contrast to the increase in the other areas indicating the movement from research to wider applications and a wider audience.

Pre 1984 systems

	A	B	C	D	E	total
1	10	0	0	5	1	16
2	24	3	28	61	14	130
3	23	0	65	22	16	126
4	1	1	4	5	1	12
	58	4	97	93	32	284

Post 1984 systems

	A	B	C	D	E	total
1	65	0	5	11	4	85
2	57	22	78	104	29	290
3	29	2	35	14	8	88
4	10	1	4	22	1	38
	161	25	122	151	42	501

Fault diagnosis is probably one of the most popular applications because many companies have a suitable fault diagnosis situation allied to the fact that it is comparatively easy to encapsulate the empirical company knowledge that is often carried around in people's heads.

The clear increase in the business sector provides further evidence of the successful transfer of expert systems from research to application. This sector, and in particular the finance domain, contains organisations which are only likely to invest in an area where they perceive opportunities not only to apply the systems themselves, but also to develop and sell their expertise through consultancies. The vast increase in the advisors is probably based on the development of relatively cheap PC-based systems which verifies the claims made by d'Agapeyeff (1984a, 1987) that systems do not have to be large and complex and that small systems can be useful.

Medical and engineering applications

It is convenient that there are equal numbers (149) of medical and engineering applications so that further analysis of these two major application domains can be undertaken.

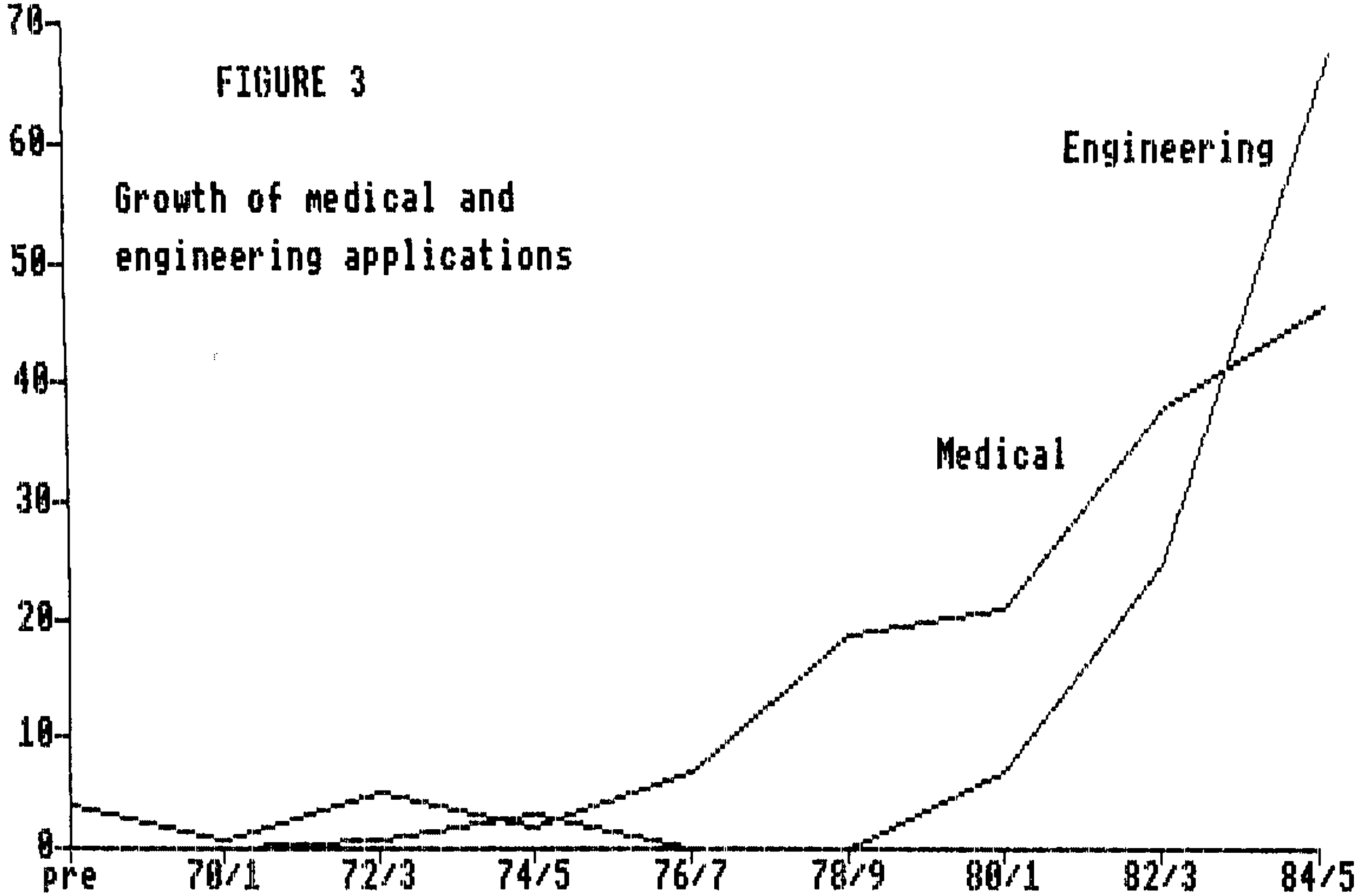
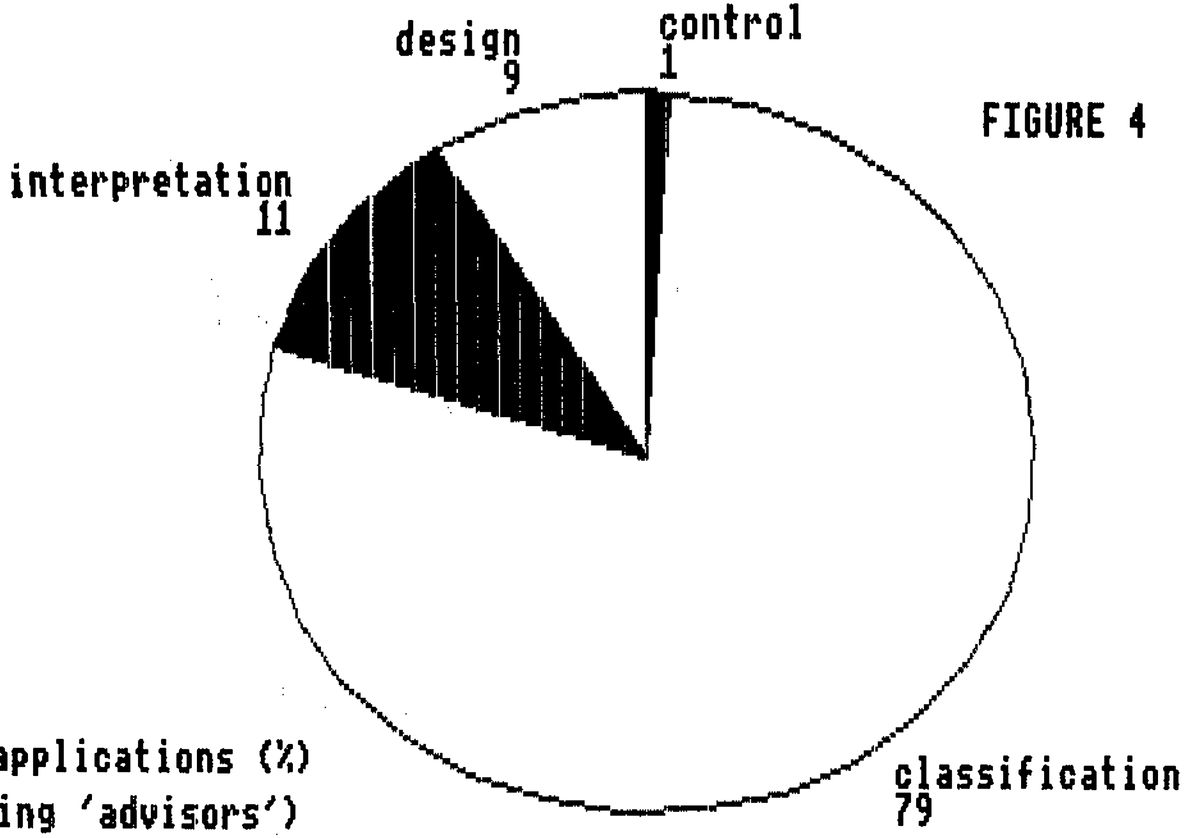


FIGURE 3
Growth of medical and
engineering applications
Engineering
Medical
70
60
50
40
30
20
10
0
pre
70/1
72/3
74/5
76/7
78/9
80/1
82/3
84/5

However, as figure 3 clearly shows, the route to the equal total is very different. In both cases there was little growth during the early 1970s, as was the case in all domains, but since the middle of that decade there has been a steady rise in the number of medical applications. In contrast, the rise in the number of engineering applications did not begin until 1980, but the growth since then has been dramatic. I suspect the reason for this difference is the fact that although medicine provided a primary area for research, the problems of turning research into application within the medical field have been more difficult than within the domain of engineering. In addition, as shown in figure 4, the majority of medical applications were generically 'classification' tasks whereas there was a much wider spread of engineering applications (figure 5).

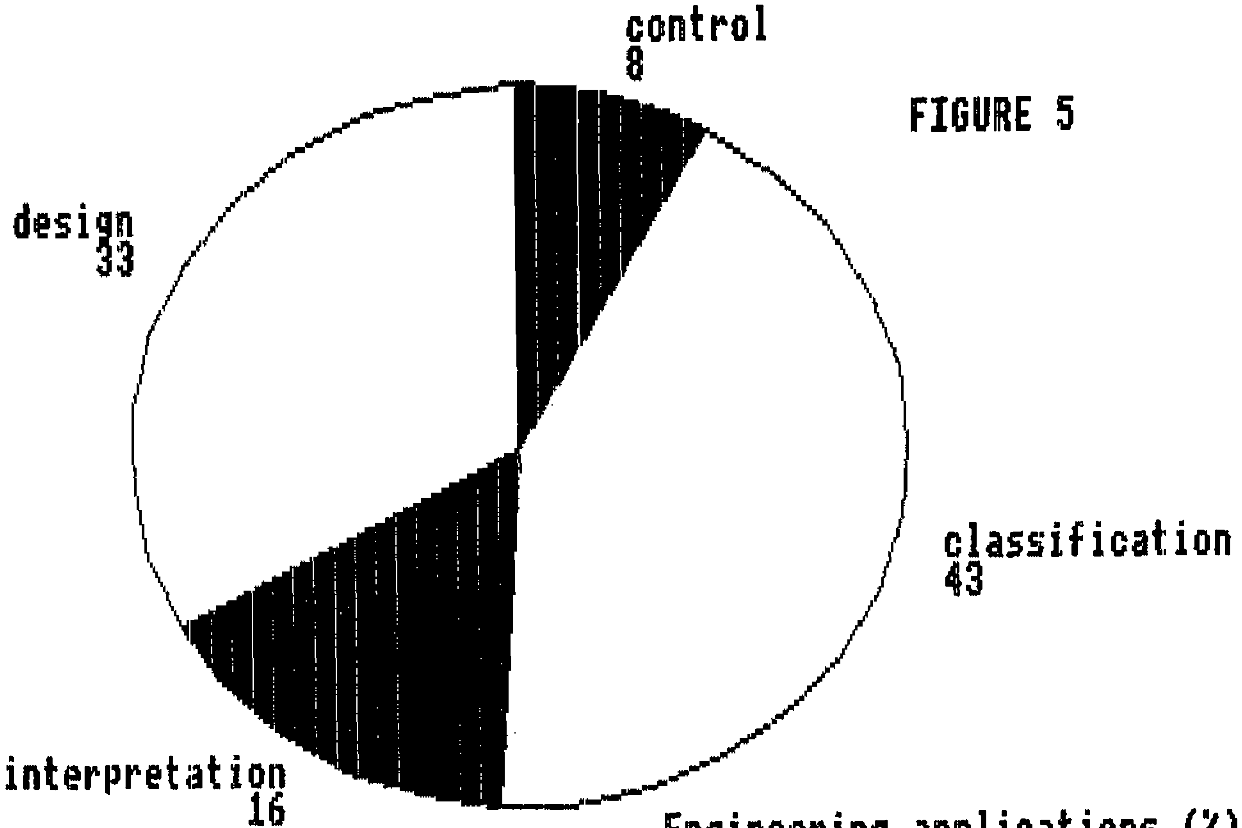
The interesting findings, illustrated below, are the increased figures for engineering applications in both UK and France. When

FIGURE 4



medical applications (%)
(excluding 'advisors')

FIGURE 5



Engineering applications (%)
(excluding 'advisors')

coupled with the fact that there has been a rapid increase in engineering applications, this could indicate that the European market is expanding.

	medical	engineering
USA	65%	52%
UK	15%	30%
France	4%	11%
Rest	16%	7%

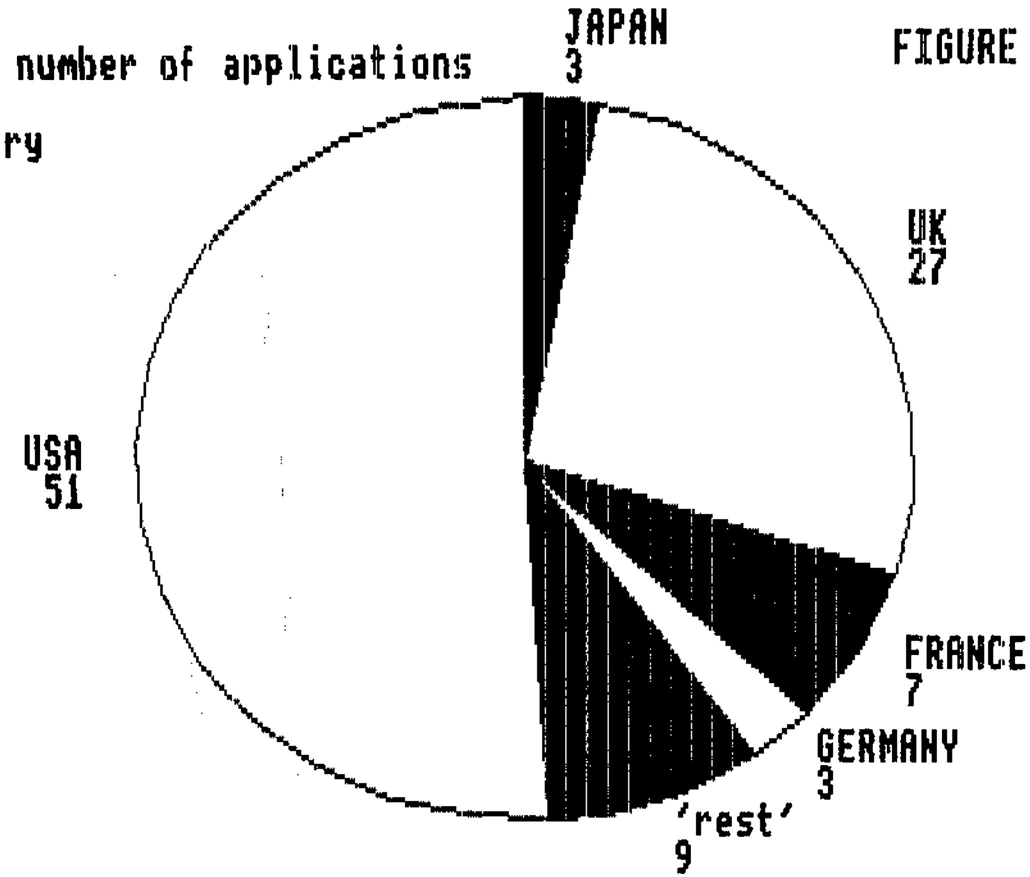
Where are the developments taking place ?

In total 26 countries are represented, but as can be seen from figure 6, the USA and UK dominate as the countries where the major developments have taken place. The countries which make up the 'Rest', mainly comprise the other European countries.

A comparison of UK and USA applications, grouped as before, shows that the USA applications are heavily biased towards engineering and science. The figures in brackets are percentages of the total UK or USA applications.

Percentage number of applications by Country

FIGURE 6



A comparison of the UK and USA domains in the business group (1), shows that there are twice as many in the UK, in percentage terms, although almost equal numerically. In the engineering group (2), by contrast, there are twice as many numerically in the USA, although equal in percentage terms.

	UK	USA
1	45 (20%)	41 (10%)
2	114 (52%)	227 (55%)
3	40 (18%)	129 (31%)
4	20 (10%)	15 (4%)
	219	412

Frost and Sullivan (1985) expect the European market to increase dramatically from $37M (1984) to $3.8B (1990) with the UK, seen as the most mature market, taking a major share of the market. This report viewed the years 1985 and 1986 as the time when the early systems tools would be replaced by more sophisticated versions. Subsequent systems would be the ones which would result in a wider market for all aspects of expert system technology. Further testimony to this huge market potential was the 700+ delegate attendance at the 1985 BCS Expert Systems International Conference and increased numbers since then.

In North America, a similar picture is painted by Hewett and Sasson (1986) who assess that spending on development is likely to exceed $400M (1986) and rise to $3B by 1992. The most active industry sectors were identified as medicine (13% of projects), computing and military (10% each), finance (6%) and communications (2%) which is very comparable to my own analysis as detailed above.

It is apparent that there is an increasing general acceptance of expert systems because of the wide variety of organisations involved in using and developing systems over a wide variety of tasks. The increasing use of personal computers, rather than specialist workstations, again supports this finding.

EXPERT SYSTEM SURVEYS

A number of surveys have been commissioned into AI, particularly expert system technology, and its effects and applications. This section will look at the main points of each one and some of their findings.

In early 1983, a limited survey (CCTA 1984) was undertaken by HM Treasury to identify and catalogue UK projects. The majority were only at the design/study stage and were being used for purposes of research, feasibility and evaluation. It was reported that a wide range of languages (BASIC, LISP, PROLOG, PASCAL, FORTH, FORTRAN, C, OPS-5, COBOL and BCPL) were being used in the development work, but only a few shells (MicroExpert, Sage and Reveal) which had not developed much at that time anyway. It is of interest to note that among the workers researching different aspects of the technology at this time (1983) were some of the present day (1988) leading UK exponents (Addis, Bramer, Bundy, Campbell, d'Agapeyeff, Fox, Forsyth, and Goodall).

A survey was conducted from August to November 1983, on behalf of the Alvey Directorate, to report on the state and nature of expert system applications in UK and to seek projects which would act as demonstrators to newcomers to the field. Evidence of the rate of change within this technology was furnished when it was acknowledged that the report could not have been written in 1982 and would be obsolete by 1986. The main finding of the report (d'Agapeyeff 1984a) was that expert systems in business are not as complex as they are perhaps made out to be. It was in this report that d'Agapeyeff found it necessary to correct the widespread impression that expert systems are inherently complex, risky and demanding. They can be much simpler and produce modest usage gains, even while still incomplete and these 'simple' expert systems can be stepping stones to more complex and ambitious applications. It was suggested that excessive ambitions have been a greater contributor to software failure than any other cause and this attitude may deflect managers from considering the potential use of expert systems in their companies.

Encouragement was forthcoming for companies considering involvement with the technology as it was described how systems can be built by self-taught teams with modest cost and little risk and yet achieve limited aims of a kind seldom achieved from conventional methods. Further practical help was also forthcoming as d'Agapeyeff established Expertech to develop and market small systems.

d'Agapeyeff (1984a), supporting the Alvey philosophy of national cooperation, also suggested the extent of secrecy adopted by user companies may go beyond the national interest. This issue, to reduce to a minimum the adverse effects of secrecy, was repeated in a later survey (d'Agapeyeff 1987). However, as a company manager in the commercial world, I would find it difficult to tell all about a development that would give my company some form of competitive edge.

Johnson (1984) identified several issues in the commercial opportunities in expert system involvement, the main consideration being the benefits to be gained from using expert systems, which, although being difficult to quantify, could manifest themselves in new levels of service, greater market share etc. The report stated that market size will be dependent upon the rate of growth of actual applications, but predicted a rise in the size of the total expert system market from $72 million (1984) to $2322 million (1990). Such a rise would not only provide investment possibilities, increase what would appear to be significant market for the supply of products and consultancy services, but also create a demand for knowledge engineering and application skills.

Predicting financial services, oil exploration and military applications as being suitable domains, Johnson (1984) suggested that early investment would be restricted to organisations providing favourable conditions, such as where the industry is wealthy and where computer technology is already well established. This report, supported the earlier d'Agapeyeff report (1984a) in predicting that small systems may well have more market significance than large ones, especially where the technical difficulties of the problem were relatively low and where the potential pay-off would be comparatively large.

Johnson (1984) and Wigg (1984) suggested that the progress of expert systems may not be quite as rapid as some forecasts because of a number of technical and social factors. Not the least of which was the fact that, despite some of d'Agapeyeff's claims about 'simple' expert systems, it remains a complex technology. Other factors such as time, cost and a lack of knowledge engineers were noted, as was the potential resistance from human experts whose livelihood may be threatened.

AI applications are becoming practical and of increasing commercial value and Wiig (1984) noted that this was particularly relevant as hardware costs are decreasing to the point where it is economically sensible to apply AI technology to broad real life applications.

A postal questionnaire survey (PACTEL 1985) received a low response from the defence sector, despite significant suspected or potential activity in the area, because defence applications were classified. Hewett and Sasson (1986) found little commercial activity in real-time systems, but suspected that the majority would actually be in military and aerospace applications. Of particular interest was the below average response from the health sector, bearing in mind the classic medical systems. This may provide further evidence of the problems of developing actual medical applications.

Although the survey indicated potential for expert systems to help the various organisations, no significant correlation was found regarding company size and position in The Times Top 100. The report suggested that there would appear to be a significant relationship between current activity and size of organisation and those firms with most familiarity with expert systems showed most belief in the potential of the technology. This potential would appear to have been well founded as, in a similar survey, PACTEL (1987) reported an encouraging level of readiness of British companies to exploit these new developments.

Although Frost and Sullivan (1985) reported that UK development would lag behind USA by up to two years, it was observed that there is widespread acceptance of their usefulness in the commercial arena and the realisation that expert systems have real business potential that differs from the traditional data processing approach. The report suggested that this acceptance will increase as a result of the greater availability of more powerful computers and the appreciation that systems can be built by domain experts, thus reducing the reliance on highly trained knowledge engineers. This has encouraged more organisations to attempt to develop systems in-house, a task which has been made easier by the development of software environments built for the naive user.

A comprehensive review of the international arena can be found in Hewett and Sasson (1986) which provides a review of the USA and Canadian activity and Hewett, Timms and d'Aumale (1986) which provides a companion report of the European scene. It was observed that the expert system industry remains schizophrenic by pointing to the rapid development of prototypes, but at the same time noting the low number of operational systems. Most of the current USA systems are at an early prototype stage with full operational status not expected until 1988. Three key trends in the market were identified;

a) the increasing use in the commercial sector of conventional languages (C, Fortran, Pascal, Ada) rather than Lisp or Prolog

b) the emergence of application-specific software

c) the increasing attention being paid to the delivery environment (more powerful and cheaper workstations)

Hewett, Timms and d'Aumale (1986) report that in terms of the number of successful operational systems, there is not much to choose between USA and Europe. Although the European market lags behind USA, the European view is more pragmatic and geared towards short term operational systems which perhaps explains why all markets, except that for expert system shells, are dominated by the USA. The greater involvement by IBM, which started in 1986, can be seen as a significant factor in the endorsement of expert systems. As world computing leader, where IBM leads others tend to have to follow. It is envisaged that by 1990 nearly all medium and large companies will have begun to explore and exploit the technology.

The 'islands' model (Johnson 1984) has held up well for a couple of years, but the islands have grown as the costs have dropped and the technology has become more established. The model still holds good, but Hewett, Timms and d'Aumale (1986) observed that the favourable environments can now be found in a wider variety of companies

> "in the 'healthy' not just the 'wealthy' industries"

This shift from 'wealthy' to 'healthy' can be attributed to a number of factors which can be summarised as a greater general awareness of the potential of the technology. However, it is worth noting that the public perception of expert systems (clever programs having a wide scope), contrasts with the long term goals of academic research and the limited benefits of existing applications. This wider awareness has meant that managers are now looking for effective solutions to problems, where previously the concept of expert systems had to be 'sold' to them. The level of technical difficulty has moved so that what was difficult is now feasible and in financial terms, the entry costs to the technology have fallen as has the size of the required pay-off. The report predicted that expert systems will now spread their applications across the range of activities of companies.

A 1984 report (Johnson 1984) identified the shortage of knowledge engineers as being the most important limiting factor in the growth of expert systems at that time. In 1986 the position had moved on so that Hewett, Timms and d'Aumale (1986) identified three separate groups of professionals working in this area, AI scientists, knowledge engineers and domain experts.

As reported in the previous chapter, the application areas are spreading widely and PACTEL (1987) reported that nearly 25% of all respondents believe that expert systems will be vital to their organisation and of these 25%, over 60% believe that the time scale will be within the next three years (by 1990).

In a qualitative update to his earlier 1984 report, d'Agapeyeff (1987) noted that there has been a substantial growth in the number of expert system projects since 1984 and that although the pace of development is increasing, the experience of operational applications is still limited and narrowly held. My own analysis reported earlier, fully supports this observation. However, d'Agapeyeff (1987) suggests that the quality of exploitation continues to be constrained by the lack of management commitment, by poor organisation, business secrecy and a fear of the cost and the nature of the technology, although the fear of the cost is not as significant as it was in 1984.

As the concept that expert systems are only for the large expensive companies has been shown to be no longer valid, d'Agapeyeff recommends that every company should consider their adoption. Although this is perhaps not a surprising comment from the managing director of a company (Expertech) which specialises in small expert systems.

All the reports mentioned in this chapter have predicted an increased expert system market. I have largely refrained from quoting predicted figures because it would be difficult to establish their accuracy and impossible to provide a standard baseline for purposes of comparison.

DECISION SUPPORT SYSTEMS

Decision making is a fact of everyday life and decisions are made at various stages of life and with varying degrees of complexity and importance. For example

shall I have tea or coffee ?

shall I get on the bus or walk ?

when I grow up do I want to be a dentist or a dustman ?

These decisions are based upon judgements which are influenced by a number of factors including the available information, experience and intuition, perceptions, emotions and common sense. The human decision making process is specific to each individual in terms of the amount of information that each person can handle and individual cognitive factors such as thinking style and speed of thought and emotional factors such as motivation.

Computers can, in general, aid in the decision making process by supplementing human failings such as;

a) forgetting things - Kleinmuntz (1968) and Feigenbaum and McCorduck (1984)

b) not taking account of negative information - Rouse (1978)

c) dealing with uncertainty and probability - Winkler (1972)

The main parts of a decision support system are some way of interrogating the data base (perhaps using a query language) and a modelling tool which also involves good file searching and data retrieval. Many systems use the label 'decision support system', a term apparently coined in the late 1970s by P.G.W. Keen. Freyenfield (1984) provides a description of decision making systems and Beerel (1987) provides the following definition.

> "a computer system designed to provide information deemed relevant to the making of a decision. Decision support systems provide support to the decision maker, but do not replace him."

The above definition could conceivably include every spreadsheet and financial modelling system, every fourth generation language or database interrogation package and all the graphics packages that quickly reduce complex data to comprehensible pictorial form. Beerel (1987) demonstrates that there are three hierarchical levels of application, a conventional database, a decision support system and an expert system. Many existing systems work by using predetermined algorithms and functions to transfer data from the user or the database. These systems work well in circumstances where data is complete or is tidily and crisply defined, but they may not be so effective where there is incomplete data or in areas where decisions depend upon 'gut feel'. Expert system technology can be of benefit in the further development of decision support systems in that they can;

a) generate systems which can cope with incomplete, less than accurate and fuzzy data.

b) provide justification for the advice or conclusions which they offer.

c) provide 'what-if' exploration facilities which are not governed by algorithmic considerations.

d) provide consistent advice which will not be prejudiced by an individual's personal priorities.

e) provide intelligent front-ends to traditional decision support systems applications.

Decision support systems are not intended to supplant the specialists, who are often under pressure to come to a decision. The idea being to provide the basis on which a specialist can make a decision, not to carry out the entire decision making process. Expert systems are also intended to help users towards a decision rather than making the decision, however this latter facility is often available.

Hayward (1984) argued that decision making in certain domains could be represented as a tree structure and coded in any programming language to search the tree. This may be more applicable in quantitative, rather than qualitative, domains. However, a tree structure is not universally accepted as the best representation as expressed by Fitter and Green (1979).

Managerial decision making is dependent upon access to information and at times, the manager must make decisions in fields in which his expertise may be limited. The 'Abilene factor', as reported by Clarkson (1986), may also play a part in group decision making. It is that groups of people tend to agree on courses of action which as individuals they know to be stupid. As it is known that humans often make decisions and then justify them afterwards, the cynic may observe that the expert system provides advice before you have made the decision and the decision support system provides supporting evidence afterwards. However, Sprague and Watson (1986) provide a clearer analysis of the difference.

Expert systems have not yet made much penetration into the decision support system applications domain, an area largely thought of as the province of corporate accountants. Other personnel are finding uses for them and several factors suggest that they may well play a major part in the future. These factors include the fact that the technology is still immature but, as the technology develops and improved human-computer interfaces are provided, along with greater acceptance of computers in management applications, the greater use of such systems will result.

Over the course of time, the boundary between decision support systems and expert system advisers is likely to become less pronounced and diminish.

COMMERCIAL COMPUTER USE

Personal computers have made computing power accessible to a wider audience and allowed those users the ability to control that power whereas their previous access was at a remote terminal. The keyword is flexibility in that the user can do what he wants when he wants. The commercial uses to which PCs have been applied revolve around decision making. These include providing advice, testing new ideas, organising thought, providing checklists, producing plans, making proposals and justifying decisions. At a tangent to all these applications is the need to share knowledge and to employ shared knowledge. The common business software (word processor, database and spreadsheet, whether separate or integrated) covers these problems. Although they are widely used they have limitations, for example, they cannot reason about empirical knowledge.

Data processing

It could be argued that every data processing system is an expert system, in that it contains knowledge. Taking for example, a payroll system, the expertise and knowledge may not be particularly encyclopaedic, but the software contains all the procedures, rules and methods for dealing with the 'out of the ordinary' situations which were formerly carried out by rows of clerks. It is probably a truism of all programs that they contain human knowledge of some form. Alty (1985a) demonstrated that the boundary between human knowledge processing and automated knowledge processing is moving so that the larger share is now occupied by automated knowledge processing.

The use of expert systems as intelligent database management systems is an example of a promising application. Databases are designed for the storage and retrieval of information, but they have not been able to apply rules based on reasoning and inferences, nor have they been able to help to extract data by users who may only have a 'fuzzy' idea of what information they seek. Such systems would require vast computing power, whereas present database systems can work very effectively on PCs. As increased computing power becomes economically available then these expert systems will become more readily available. Expert systems which could update their knowledge base based on experience would be a valuable step forward. Paice (1986) discusses the relevance of expert system technology to information retrieval which is considered in more detail in chapter twenty nine.

The first commercial interest in expert systems began at the beginning of the eighties with the stimulus coming from activity in American universities (especially Stanford and CMU) and from the Japanese Fifth Generation initiative. Ishizuka (1984) reported that the Japanese work at that time was directed at medical diagnosis, plant control, CAD, image processing and management and office systems.

As commercial organisations rely widely on conventional computing software, it has been suggested by Jones and Davies (1986) that the successful systems will be those that easily integrate into existing software to provide an evolutionary pathway and that a likely future scenario is for expert system technology to be embedded in other software.

Expert systems differ from conventional data processing systems in that they involve such ideas as symbolic representation, symbolic inference and heuristic search. In a conventional system the rules by which the program reaches a conclusion are implicit in the program code. In an expert system, such rules are made explicit and stored along with facts about the problem area in the knowledge base. The program then only needs to know the general strategies for applying these rules and using them in combination to infer new conclusions from existing knowledge. Hence the system builder can represent both precise arithmetic or logical relationships in his rule-base AND ad hoc heuristics. A further possible development is that if the rules can be made probabilistic rather than certain, this would enable the system to infer conclusions with an associated degree of confidence, enabling the system to offer a range of alternative solutions ranked in order of confidence.

The ALVEY angle

The Alvey Project (Alvey 1982) provided the possibilities for collaborative research and possibly a change in the base of our national commercial activity in favour of a more international approach. The Alvey Project was set up in 1983 as a response to the Japanese 'Fifth Generation' computer project, variously summarised by Fuchi (1983), SERC (1983), Ishizuka (1984), Stewart (1985) and Durant (1987). Alvey was intended as a five year programme with a budget of £350 million. Research would be divided into four main areas;

Man Machine Interface

Software engineering

Very Large Scale Integration

Intelligent Knowledge Based Systems

It was realised at the outset that the IKBS programme would have to be viewed as a ten year, rather than a five year, programme. The IKBS programme covered 100 projects, ranging from 'show-me' projects to short-term and long-term projects, authorised at a total cost of £28 million. This IKBS funding, provided by the Alvey Directorate and industry was divided as follows (Shorter 1987);

25% Expert system projects
20% development of software tools
20% research issues
20% IKBS demonstrators
15% awareness programme

As part of the Alvey Awareness programme in IKBS, and almost as an afterthought, a number of 'Community Clubs' have been established to focus attention on expert system technology within vertical markets. The aim being to encourage cooperation and collaboration on the research and development of a system within the club's domain. It was envisaged that, for example, a general loan processing system could be developed by ALFEX and then each member could tailor the system to their specific needs. Funding comes from the members of the club paying an annual subscription which is matched by a grant from Alvey. Appendix 3 provides details of the membership of the Clubs. The 'club' concept has been carried over into the use of expert systems in training by the formation of a 'Training Club' which is discussed in chapter twenty seven.

Shorter (1987) reported that not all the Alvey projects aimed to complete within the life of the programme, but at least one feasibility study has been sufficiently successful for further development to take place in commercial secrecy and with no further Alvey funding. The Clubs have been successful in defining the needs of the expert system user community, developing ideas about applications and allowing suppliers to establish new products and services. The Alvey programme as a whole was aimed at developing a collaborative spirit, but as suggested earlier by d'Agapeyeff (1984a), companies who, a couple of years ago, were open about their involvement in expert systems are now putting up the shutters as they see their developments as giving them a competitive edge. As Wardropper (1986) noted

> "AI is coming out of the academic confines, but seems to be going straight into a commercial closet instead."

There has been much debate about the successor to Alvey, but the Government seem reluctant to provide any more money and so industry must foot the bill. An example of a possible way forward is a project known as TAURUS which is drawing its members from two of the IKBS Clubs (ALFEX and ARIES) with Stage 1 funding (£125000) being provided entirely by the club members. The project is aimed at developing an architecture to enable separate systems to cooperate with each other.

However this transition seems to be accompanied by a corresponding increase in the marketing hype that surrounds each supplier's product. Everybody seems to be pushing their 'intelligent' products, but 'intelligent' in their eyes only. A study of the computing press will show that many products contain the words 'expert' or 'intelligent', although it is debatable how many of them would justify such a label. It is a criticism of 'intelligent' software in general, not only of expert systems, that it takes in symbols (of which it has no comprehension), manipulates them according to its 'rules' and outputs the answer (again of which it has no comprehension). This is a fair criticism, but takes no account of user expectations. The computer-user community may have little or no idea what the technology is and, in many cases, does not really care how it functions. As long as the software does the job that it was bought to do then that would be quite satisfactory.

If the object of the exercise is to produce a reliable working system then the developers are going to employ proven conventional techniques. However, the software technology is still developing and along with hardware advances, there is likely to be a widespread development of the uses for such systems by the end of this century. The attitude of the commercial world has changed in that it is now tending to look to the technology as a provider of system development aids.

Expert systems were designed for solving problems where;

a) experience and expertise is involved

b) the problem cannot be modelled quantitatively or solved by a mathematical algorithm

c) knowledge is not accessible or is subjective or experiential

d) situations require the use of judgement.

Nevertheless expert systems are not systems for every application including solving all of the world's ills, they are after all only computer programs and there is no inherent magic. The answer cannot be encapsulated as 42 (Adams 1978). They do, however, offer new methods of solving old problems, a way of encapsulating perishable expertise, of distributing such expertise, converting knowledge into a possible competitive edge and providing a new perspective on some of the problems of the increasingly complex business world.

Johnson (1984) explains the shortage of large operational expert systems (eg XCON) as a function of the time needed to bring such complex systems into operation and the lack of appropriate human skills and experience.

The expert system market is still immature and there is not yet a commitment from UK industry to significant investment. The market is not going to take off until users can buy ready-made software that doesn't require programming expertise to alter. There are grounds for believing that the initial purchases of expert systems were made by groups or individuals who wanted to experiment with and learn about the technology. The ability to assess and evaluate the technology was a prime reason for the production of the Alvey/NCC Starter Pack (NCC 1985). The situation has changed so that present purchases are made by people who are generally not as interested in the technology per se, but just want the advice that the system provides.

Companies are realising that knowledge is a tangible asset to their organisation and expert systems can be a good means of pooling resources, particularly if the knowledge of several experts can be encapsulated into the system. Although it may be difficult to show it on a balance sheet, as Banks (1986) argued, it is only possible to quantify the value of knowledge or the cost of replacing or acquiring it, probably when an expensive mistake has already been made. This realisation has considerable training implications which are discussed in chapter twenty seven.

The actual commercial use of expert systems in particular domains may reflect the interests of the developers or those providing the funding rather than the fundamental properties of the problem domain or of the particular systems. It is probably not surprising to find a variety of roles for such systems within the field of computing and telecommunications. Expert systems offer such companies a huge potential as the complexity of their product lines make them ideally suited to fault diagnosis and configuration.

In the past, it didn't take an expert to know what was inside a software package, but as packages become more complex then the need for expert help becomes more crucial. This can take the form of an expert system acting as an intelligent front-end (IFE) to the package or as a decision support system. Examples of such systems from a variety of domains include;

CHEMG - chemistry

CPSFE - contour plotting

DIAEX - image processing

ELAS - oil drilling

GEOLO - oil exploration

GLIM - statistics

NTGAS - data retrieval

XSEL - part of the XCON system

The difference between the two can be represented as the difference between having a human adviser (IFE) and reading a book, but still having to make the decisions yourself based on what you have read. Expert systems can operate over a range from the level of the book to the level of the expert human adviser.

FINANCIAL IMPLICATIONS AND APPLICATIONS

There are two issues which will be discussed in this chapter; the use of expert systems within the financial domain, which is discussed in further detail by Guilfoyle and Jeffcoate (1987) and the cost benefit analysis of using expert systems in any domain.

Simons (1983a) identifies the following features of a financial problem which make it amenable to expert system treatment;

a) where no suitable algorithmic approach exists

b) where the costs of bad decision making are high

c) where non-experts are likely to make bad decisions

d) where the problem and the knowledge domain are static and not time dependent

e) where the problem can be solved by experts

f) where the political climate is suitable for its introduction

In financial services, individuals are often required to make value judgements based on an assessment of facts against a set of rules. Dungan and Chandler (1985), Accountants Magazine (1985) and Mathieson (1986) provide analyses of why expert systems are suited for accounting applications;

a) internal use for greater efficiency

b) selling their knowledge to the outside world

c) providing a consultancy service for companies wanting to set up their own systems

An expert system could find use as an information filter, particularly for dealers at times of pressure. This is an application that will increase as computer technology increases both the speed of presentation and the amount of available information. There is also money to be made as demonstrated by TADIS, a system developed by Data Logic to advise dealers on fluctuating foreign exchange markets. Over a three week period, it was reported by Hockaday (1986) as having performed better than City institutions. Johnston (1985c) reported that of the $15 million profit from arbitrage, at least $1 million was directly attributable to the use of expert systems.

Systems can be used to buy and sell shares when they reach predetermined levels. Under normal circumstances, financial markets move up and down, but a 'crash' situation may arise as a result of an 'intelligent' system selling shares when they reach a fixed arbitrary level. This situation may be compounded if there are several of these systems in action simultaneously. The consequences may be dire. The October 1987 share price fall on Wall Street which started similar falls across the globe was blamed upon the action of such 'intelligent investors'. It is difficult to assess the truth of this statement and even more difficult to quantify the share of the blame that should be apportioned to expert systems. However, Essinger (1988) concluded that computer-assisted trading was not responsible for the crash, but may have increased the volatility of an already falling market.

Hewett and Sasson (1986) report that expert system development is shrouded in secrecy, either because the companies are keen to preserve any competitive edge which systems may have given them or because they are not doing very much. They believe that the latter is the case, UK development being about two years behind USA, but expect developments to take place. They report similarly about the insurance sector. Banks and other financial institutions are not the sort of organisations to spend the vast sums of money that they have on research and development into this technology unless they were sure of the benefits. Further information on the applications of expert systems in this domain can be found in Ernst (1988).

Cost-benefit analysis

The routine use of operational systems will be limited to examples where they can show a good cost-benefit analysis. The applications will spread as their credibility increases and as the cost comes down, then the cost-benefit analysis will admit more examples into the users club. d'Agapeyeff (1984b) suggested that there is the problem that it may only be possible to produce a cost-benefit analysis for management afterwards. Early efforts are likely to produce qualitative, rather than quantitative, improvements.

For a system to be successful it could demonstrate a new research technique or add to the general fund of AI-related knowledge, but for a system to be commercially successful, it must be put to use on a daily basis and show that it is the most cost effective way of tackling the particular problem.

Commercial products fall between expensive turnkey systems or shells at the low end of the market. At the completion of a nine month study of commercial applications, Doris Kovic, a consultant with Macintosh, reported in Mill (1986), found that few software products fall between the low cost shells and the £100,000+ systems. This situation will change as the turnkey systems come down in price and become more general off-the-shelf systems and the shells become more sophisticated. At present there appears to be an inverse relationship between cost and flexibility. Although most reports predict an increase in the expert system market, it is not expected to take off until users can purchase ready-made applications that do not need programming expertise to alter rather than either buying a shell and putting in the knowledge base themselves or using LISP or PROLOG and employing expensive hardware and programmers.

Expert systems were identified as solutions in search of a problem and the prime concern of commercial users is that of applications. The technology may be interesting or even highly advanced, but until its value can be quantified, then the businessman will not be particularly interested in anything more than initial evaluation and awareness. The organisations that are actually developing systems usually had some specific project in mind, the remainder were evaluating the technology in an attempt to keep up with their competitors. Additionally the development seemed to be the province of the larger firms, those who could afford a research and development department or those who could stand the time lag and associated cash flow deficiency between system design and the development of a real working system.

LEGAL IMPLICATIONS AND APPLICATIONS

This chapter discusses the use of expert systems within the domain of law and also the legal implications of the wider use of expert systems.

Law must be a particularly difficult domain for expert systems to enter. Interpretation of regulations may be feasible, but much of the law is concerned with interpretations, flexibility, precedents and creating precedents. Adam and Taylor (1986) noted that

> "the law is a dynamic process where legal rules are never clear. The rule of law which is deduced from one case and applied to a second case, cannot be regarded as fixed, as the rule is changed and remade in the process and there are always fundamental ambiguities. The judge in a new case is not bound by a rule of law made by a prior judge, but may emphasise other facts in

formulating a new and different rule."

It is unreasonable to expect an expert system to cope with such circumstances, particularly as Gardner (1987) observes that statements of the law are written in natural language and furthermore, legal arguments are often arguments about what the language means or ought to mean. Additionally, Broomfield (1987) argued that at a deep level, PROLOG programs do not 'think' like human beings and more importantly, that they are not infallible.

In the context of a criminal law trial, it is necessary for the jury to weigh the evidence for and against the defendant. This evidence may include information of unknown reliability, it may be in partial conflict and it may be ambiguous or otherwise imprecise.

There are two problems for those who wish to build expert systems in law, firstly to find areas where there is a broad band of consensus over the knowledge to be incorporated into the system. Legal knowledge, as applied by lawyers, is more strategic than rule-based and on that basis would be impossible to incorporate into an expert system. Secondly, it is necessary to build systems which are sufficiently usable, although this must be a problem for developers of systems in any domain.

The Alvey project to formalise the British Nationality Act (Kowalski and Sergot 1985) was based on the premise that much human knowledge and belief can be usefully formulated and analysed using logic and such analysis can help clarify and simplify legislation. This premise was strongly refuted by Leith (1986). The British Nationality Act was chosen because it was a small, self contained piece of legislation. The project workers accepted that they would not produce an automatic system to determine questions of citizenship, because provisions laid down in the legislation are only one source of the law. However, Leith (1986) questioned as to whether the project workers understood the law sufficiently well in order to be able to complete the system.

Keeping up to date with old and new case law must be a nightmare and traditional computer techniques for search and retrieval tasks have been used for many years. 'Intelligent' search techniques could well find applications in the legal domain eg Hafner (1981). Feinstein and Siems (1985) describe a system, EDAAS, which screens requests for information under the USA Freedom of Information Act.

There has not been much legal activity as a result of the use of expert systems, largely because of the relatively limited numbers of systems that are in operation. As the number of advisory expert systems increases, then the potential for legal action arising from their use, and their misuse, also increases. The problem for the Courts is how to handle the inevitable controversy that will accompany this proliferation of cases. The law will take its slow evolutionary path as precedent is created and amended in the light of both individual circumstances and the development of more advanced systems.

No expert system is foolproof and a dangerous assumption for any user to make is that every piece of advice provided by the system is exact and correct. This is not so, it is a piece of advice to be heeded or discarded as the user sees fit. Since some expert systems work with uncertain knowledge, the conclusions that they reach must also be uncertain. Murphy's Law applies to computer software as much as to any other commodity and errors are still going to be made. The software may malfunction, but this may mean that it does not work as it was intended or that it does work correctly, but was misused. Lucash (1987) identified five sources of malfunction

a) basic design

b) programming errors

c) use of incorrect or faulty rules

d) improper implementation

e) improper selection or misuse

Smith and Baker (1983) questioned at what point should a system be released, when, by the nature of changes in knowledge, there will never be a final version. It was further questioned as to how subsequent changes and alterations could be managed if and when the system was released.

The software producers have realised the problems in this area, hence the rider or disclaimer that usually accompanies any output from a system. For the user it means that expert systems should not be uncritically believed. Although this is widely accepted, the legal implications have not yet been fully evaluated. Arthur and Watts (1986) report that this problem has been recognised by the Royal College of Surgeons and ICL. The latter market MEDICL, a system for diagnosing abdominal pains, which contains a built-in clause which forces the user to accept responsibility for the patient before it continues with its diagnosis. Such action is aimed to protect both ICL and the product. The inclusion of a disclaimer in both the documentation and the program forces the responsibility for the use or misuse of the expert system advice onto the human user. This is not an unreasonable assertion to make in the normal course of events, as it is likely that the courts will trace the liability back to a human source. In many cases this will usually be the 'expert' who provided the rules for the system in the first place. As the widespread use of expert systems can quickly promulgate any 'mistakes' over a wide area, this is an area where it would appear that the experts would need not only to check and double check their facts and rules, but also seek means of limiting their legal liability. However, an extra level of complexity is introduced when considering the use of real-time expert systems. Many such systems do not interact with humans, but with external control devices. The legal position is unclear if, for example, an expert system orders a particular action which results in a catastrophe. Do you sue the plant owners, the plant controllers, the knowledge engineers, the domain experts ?

Zeide and Liebowitz (1987) discuss some of the USA legal issues relating to using expert systems and identify as one of the prime legal issues, the different way that the courts treat liability for injury resulting from products or from services. They also provide two situations which encapsulate some of the problems to be faced by the legal experts;

> "Case 1: patient B describes her symptoms to Dr Y who treats her without resorting to an expert system, even though one was available. Dr Y misdiagnoses and patient B dies.
>
> Case 2: patient A describes his symptoms to Dr X who consults an expert system and bases his treatment upon the recommendations

> of the expert system. The expert
> system has misdiagnosed and patient
> A dies."

In both cases legal action could be taken against the doctors, but the debate would centre around who is to blame. Legal proceedings could be taken against a doctor who has accepted diagnostic advice from an expert system which was later found to be false and as a result of following the advice of the system, the patient died. Short answers are not appropriate, not only because of individual differences from one case to another, but also because expert systems are both products and services. The following questions highlight some of the many legal wrinkles and problems;

Dr Y normally consulted an expert systems, why didn't she on this occasion ?
Why did Dr X adhere strictly to the recommendation of the expert system ?
The expert system contained a disclaimer, but what is the legality of such a disclaimer ?

How would the situation be affected if the diagnosis was made by an expert system, but the expert system was used not by a doctor, but by a nurse or a layman ?
What would happen if the damage was caused by the user himself ?

Arthur and Watts (1986) reported on a further legal problem, identified by The Royal College of Surgeons, if computerised diagnostic systems become available on public viewdata services. There would not appear to be a problem if used by a qualified doctor but there could be a problem if lay people started using them. At the moment this will not happen because the systems are not available on, for example, Prestel. If it does and you, the layman, or possibly a qualified practitioner, use the system with dire consequences, then, assuming that you are at least still alive, who do you sue ? The legal profession are likely to have a field day untangling all the legal and medical spaghetti. At the moment, nobody has given the prospect much thought. One possible solution is to include a Government Health warning so that if people use it and kill themselves then it would be their fault. There is a standard 'duty of care' that is applied throughout the medical profession to the effect that if you apply a bandage or diagnose a disease that you do it correctly. The patient applying self-medication would have the duty of care to see that it was done correctly. However, if it could be proved that the system was at fault because it didn't ask you about a particular condition then........

Lucash (1987) argued that present law is insufficient to deal with the applications of expert systems. This is certainly an area where many legal, moral and ethical questions will need to be tested and clarified before expert systems become more widespread.

MEDICAL APPLICATIONS

There is a relatively long history of computer involvement in medicine and information about the medical applications of expert systems has been easier to find than in almost any other field. Rogers (1979) noted that there are several reasons for this, especially the fact that publicity has not been inhibited by the commercial pressure to keep the system secret and the availability of research funds.

Computers have been used in the field of medicine for many years. Simons (1983) sees computers having the following inherent capabilities well suited to medical problem solving;

a) the ability to store large quantities of data, without distortion, over a long period of time

b) the ability to recall data exactly as stored

c) the ability to perform complex logical and mathematical operations at speed

d) the ability to display many diagnostic possibilities in an orderly fashion

The initial medical use of computers was to juggle figures to some end-purpose and such an algorithmic approach is satisfactory and if there is an algorithm, then it should be used. In the early days the programs were based upon mathematical or statistical work and were used for medical decision making. In an early survey, Rogers (1979) found that 60% of all the diagnostic studies used an algorithm based on Bayes' Theorem.

Often, the situation is too complex for such an approach and the use of AI techniques emerged during the 1970s as a response to several simultaneous, but unconnected, needs, opportunities and interests. However, medicine is an area where the development of operational expert systems has been slow because of a number of factors. The clinical tests and trials, necessary before they are allowed into routine use, is a long term process. Hewett, Timms and d'Aumale (1986) reported that the final medical trials of a system were completed in UK fourteen years after the first results were published.

In addition, persuading doctors to accept and use them as routine tools is, as with most innovations, not a foregone conclusion and consideration needs to be given to the psychological elements of using computers as consultants.

MYCIN

MYCIN (Shortliffe 1976) is a well-documented diagnostic medical expert system, although it has only been used experimentally and has not found regular use on the wards. Further suggestions were made as to its continued development. One idea was to use MYCIN's rules in a teaching situation. The knowledge in MYCIN was found to be too narrow to be used to teach a student to be a primary diagnostician. It was further noted by

Clancey and Letsinger (1981) that simply adding more knowledge to broaden the scope of the system would not result in a successful tutorial program. Additionally Clancey (1986) noted that MYCIN lacks the sophisticated explanation facility needed in a teaching role. NEOMYCIN was developed as a consultation system which uses the knowledge base of MYCIN in a teaching program, GUIDON (Clancey 1982). Clancey and Letsinger (1981) believed that

> "NEOMYCIN was the first attempt to formalise a runnable psychological model of diagnostic strategy which can be presented to a student."

A psychological model of problem solving needs to be incorporated into any system which attempts to teach diagnostic strategy. This was necessary because Clancey (1986) found that the expert doesn't organise or use knowledge in the same way as the program.

A problem for any system working within a medical domain is that not only are patients unique, but treatment regimes differ and the diseases themselves are changing. Hence, medical systems, to be continually effective, must be capable of adapting, which is comparable to the need for adaptation to the user-model incorporated in an educational system.

TEIRESIAS (Davis and Buchanan 1977) works in association with MYCIN, collecting new rules from the expert, checking the consistency of the new rules with the existing rule-set. Any of the existing rules which, in the light of the new rule, appear inconsistent or inadequate are highlighted.

During this century, medicine has developed to the extent that no one doctor can be an expert in all fields, hence doctors become specialists in one field of medicine or another. Modern medicine has become very complex, the amount of data available to the doctor has increased dramatically (partly due to IT), but at the same time, the cognitive capability of the doctor is a relatively fixed quantity. As the problem had arisen partly as a result of IT, medicine has looked to the technology to aid in solving the problem. Society demands higher and higher levels of health care and service, but most diagnostic decisions are based on rapid judgements of the patient and relying upon the doctor's memory and experience. This increase in demand will continue, Schwartz (1970) predicts that the health care of 2000 will be totally different from that of today and that the exploitation of the computer will be involved.

Alexander (1987) identified three kinds of reasoning and explanation required by medical expert systems, for diagnosis, underlying causation and for remedies. However, medical decision making is not just about diagnosis. McSharry and Fullerton (1985) showed that patient management comprises several mutually dependent activities all of which require decision making;

a) what investigations and tests to use

b) making the diagnosis

c) treatment selection

d) prediction of the prognosis

In any consultation with a doctor, the doctor will follow his 'rules' and perhaps recommend a specific course of treatment unless there are particular grounds for not doing so. These grounds may be based on medical factors or on more subjective personal factors. The expert system will be able to match the doctor's course of action up to the point where the doctor's own subjective judgements come into play. The medical system is not intended to replace the doctor, but to perhaps act as a collator, looking for trends and picking out obscure things that the doctor might have missed or not known about. Brown (1985) showed that diagnostic or management mistakes are usually the result of such errors or omissions.

Medical applications of expert systems are increasing, for example, in August 1986, ICL

released MEDICL which helps doctors diagnose abdominal pain. St Thomas's Hospital has been working on one to diagnose diabetes and in Australia the Garvan Institute of Medical Research has a system doing thyroid test diagnosis in everyday use with a reported 96% correct rate.
However Arthur (1986) and Clark (1986) speculate that the widespread use of medical expert systems will not take off until they can be speech driven. Student doctors using the MEDICL system found that their accuracy increased from 40% towards 60% (the level of a consultant). The system used a protocol sheet to describe the symptoms and it is possible that the use of the sheet, as an aide-memoire, helped just as much as the computer.

Medical diagnosis situations are often complicated by the fact that more than one disease may be present. Clancey (1983) has suggested a need for the general diagnostic strategies to be made explicit and kept separate from the more specific information needed for the diagnosis of particular diseases.

There is growing recognition that expert systems can be very useful diagnostic aids. The accuracy of a computer-based diagnostic system is dependent upon the complexity of the diagnostic task, the amount of data in the knowledge base and the method used by the system. However, there is a wider market for such systems, particularly in the Third World where there are medical problems but also a lack of doctors. Expert systems can be used to disseminate knowledge so that if human skill can be encapsulated in software to run on cheap and portable micros then the resulting computer system can be duplicated at virtually no cost for the software and just the hardware costs to find. The resulting system could provide assistance to millions of users.

A further use of expert systems that has found medical applications is in the discovery and refinement of knowledge.

Forsyth (1984a) developed BEAGLE, a knowledge induction system, which was tested on a file of 100 heart attack patients. A number of measurements were made as the patients entered the intensive care unit and their progress was then monitored. After 500 generations of rules, the system came up with a rule which was 81% accurate. Initially the doctors were sceptical about this finding, but on later reflection it was discovered that there were good medical grounds for the rule.

Similarly, there are reports (Expert Systems 1985) on the development of two Expert-Ease applications. CHEST PAINS, a system to help diagnose, with accuracy, potential heart problems and an unnamed system which helps to predict the likelihood of a clot forming in the left ventricle (such clots can lead to embolisms which can be fatal). The developer described the process of learning to use the program as somewhat painful, but now says that the system is incredibly simple.

In the KARDIO-E project (Lavrac et al 1985), it was found that the knowledge relating to the characteristic ECG features and their diagnostic parameters could not be found explicitly in the medical literature. In a similar vein, at the end of 1986 the Imperial Cancer Research Fund, allied with Oxford University Press, released a system (the Oxford System of Medicine) for doctors to use on an everyday basis. It is a prototype system with facilities to store patient records and medical text like an electronic book, to assist with diagnosis and to explain how it arrived at its decision.

A 1985 study chaired by Professor Neil McIntyre of the Royal Free Hospital, reported by Watts (1986), has shown that a computer-based diagnosis system could improve performance and save money. The study looked at eight hospitals using a system to help diagnose acute abdominal pain involving over 17,000 patients. As a result of using the system, the number of unnecessary operations and the number of patients admitted was halved. Additionally, 33 fewer patients died than would normally be expected and the combined savings amounted to over 4,000 bednights a year. The report makes clear that the computer in no way replaces the doctor and concluded that

> "the work demonstrates beyond any reasonable doubt that the system works reliably and effectively......wide use of the system within the NHS would save between £20 million and £25 million in recurrent costs and £5 million in direct costs per annum."

Situations where the expert system assists a human operator appear to be productive. A system in use at a London hospital is helping radiographers understand pictures from a brain scanner and it is claimed that through using the system they are able to make more accurate diagnoses than before. (Expert Systems in British Industry, Alvey Video. Open University)

Time was noted by Gotts et al (1984), as a factor to consider in any medical situation as the condition of the patient will vary. It may be a sudden or gradual change, be reversible, cyclic or irreversible or of a long or short term nature. It may even be static and that in itself may be a significant condition. In the diagnosis of a disease, the course of events is often a characteristic feature. In some cases it is sufficient to know that A preceded B, in other cases it is necessary to know the time interval between A and B. A medical expert system must have the capacity to represent changes in the patient's state over time and to take due account of such changes.

The use of any medical system brings with it associated legal problems of who is responsible if the machine makes a wrong diagnosis. Systems often get around this problem by offering a number of weighted possibilities so that they only aid the decision making process rather than making the judgement. There must be the usual caveats regarding accuracy where probability is concerned. Hence the final word on the medical applications of expert systems comes from Gotts et al (1984)

> "Medical expert systems should err on the side of caution and they ought to make it possible for the user to make an informed decision rather than make the final decision themselves."

ENGINEERING AND OTHER APPLICATIONS

One of the main reasons that engineering applications have led the way in the development of expert systems is that only oil companies, such as Exxon and Shell were capable of not only providing the huge investment that was necessary to develop a truly commercial system, but of also being able to withstand any losses that may occur. The fact that they were prepared to invest so heavily in the expert systems market must be seen against the vast losses, reported by Else (1985), made by Exxon when it attempted to diversify into the electronic office market. Some firms discovered the level of required funding the hard way. In 1983, Racal and Norsk Data set up Racal Expert Systems, a company aimed to sell expert systems to the oil field exploration industry. Mill (1985) reported that after a year spending £1 million and still without a commercial product, the company was disbanded.

The publicised expert systems in oil

exploration (eg DIPMETER ADVISOR, DRILLING ADVISOR) have, as observed by Johnson (1984), been large, ambitious projects which have proved to be difficult to put into operation in the field. Further overviews of expert systems applications in engineering are provided by Rychener (1985) and Sriram and Rychener (1986).

Manufacturing applications

Knasel (1986) predicted that by 1990 manufacturing use of AI will grow from 10% of all AI use, to 30%-50%, provided that;

real-time control applications emerge,

no AI skill is needed to use or install,

the system runs fast enough on a standard machine

the software license costs no more than $100 per installation.

However, I consider these to be optimistic predictions.

Kempf (1984), acknowledging the enormous supporting role that conventional data processing plays within the manufacturing domain, argues that in manufacturing, conventional and AI computing techniques are complementary because they attack different classes of problems. O'Connor (1984) identified the manufacturing environment, where there is constant change due to increase and cancellation of orders, changes to business input and changing demand requiring variations to product lines as being one area not amenable to traditional algorithmic solutions and therefore a suitable target for expert systems development.

In many areas of manufacturing industry there is the vital task of selecting a mixture of component parts and creating a saleable product. In a highly competitive market, the knowledge and skill of the product formulator will be critical. The PFES (Product Formulation Expert System) project, an Alvey demonstrator project, tackled the problems in this domain (Alvey Mailshot 1987).

Developments in production planning, control and expert systems are reported by Oliff (1988).

Real-time and military systems

Real-time expert systems are more complex than other expert systems because of the constantly changing nature of their data input. Computer controllers have uses ranging in size from the small home use to the large commercial application. Increasingly these controllers are taking on more and more complex functions. This complexity is increasing not only in the number of functions that are under computer control, but also in the number of factors and the level of their complexity that is required to make a control decision. The potential applications of expert systems in these cases seems to lie at the periphery of the control process itself. For example, experiment and test planning, data interpretation, equipment tuning and a variety of advisors. Some applications however are at the core and are capable of overall real-time control. An analysis of the problems of implementing a real-time system, with particular emphasis on HEXSCON, is found in Wright et al (1986) and YORKTOWN ES/MVS in Ennis et al (1986). Turner (1986) provides a discussion of the considerations involved in the design of expert systems for time-critical, as opposed to time-varying applications.

Time is a crucial factor in any real-time system, but there is a wide divergence from one application to another depending upon the specific task and circumstances. When required to do so, HEXSCON (Wright et al 1986) can make responses in 10ms, whereas because of the different nature of the task, LINKMAN can take its time and arrive at a decision after 10 minutes. As the time factor becomes more critical, there is greater need for the system builder to include a system for ordering the tasks to be undertaken. Determining priorities in a complex situation may actually take longer than the task itself and it would be of little use if the system shut the stable door after the horse had gone. Banks (1986) noted that where the time element is critical, it is essential to obtain the best possible solution within the defined time limit even though the solution may not be the most complete answer.

In complex situations where vast amounts of data may be coming in every second, it requires a highly trained engineer to control the plant. Typically about 3000 signals need to be monitored on a North Sea oil platform. Under these routine operations, or particularly in the case of an emergency, the engineer could have cognitive or information overload and may not know which piece of data to respond to first. This may well be a critical decision. A potential solution is for a system, such as ESCORT, to advise operators of the relevant priorities. Further help might be at hand in the form of a real time system which would physically control the plant through a system of sensors and control devices. Hewett and Sasson (1986) report on the use of PICON at several oil refining installations (Texaco, Exxon). The pay-off for such companies would be substantial because of the vast cost of running the refining process.

Although an overview of the military use of expert systems is given in Stewart (1986), many applications within this domain have been hidden behind a cloak of secrecy (many of the American DARPA projects come into this category). Hence an accurate assessment of the type and level of activity within the domain of defence and aerospace is impossible because of the sensitivity of the work.

Scientific and other uses

Computer programs have been widely used, over a number of years, in the field of sciences, particularly for 'number crunching' applications. MACSYMA (Moses 1971) is an early example of a large system that is used to assist scientists and mathematicians in tackling mathematical problems. It accepts symbolic inputs and gives symbolic outputs in addition to its algebraic manipulation skills. MACSYMA is available via a network to many hundreds of US researchers who use it on a daily basis and Simons (1983a) provides a list of the range of applications.

The geologist working with the PROSPECTOR system prepares a model as an inference network. The system hit the headlines in 1982 when it discovered a multi-million dollar molybdenum deposit that the expert geologists had missed. That success had a two-fold effect in that it gave expert systems research an undoubted boost, but raised public expectations of such systems to impractical levels.

WHEAT COUNSELLOR, an agricultural system, is the first expert system to be available on videotext, using a knowledge base held on a central computer. This may provide an insight into future possible applications.

Stefik and deKleer (1983) pointed out the increasing scope for expert systems within design applications by using the expert system technology to reduce the complexity of the task to manageable proportions. Coyne et al (1988) discuss the uses of expert systems for design applications and the advantages of expert systems over conventional CAD packages are further discussed by Simons (1983a) and Simmons (1984).

An interesting application (Practical Computing 1987) involves the Devon and Cornwall Constabulary who have been experimenting with expert systems for a number of years and have produced a 150-rule burglary system, based on the modus operandi of the burglar. They have found that the whole process of developing the system has had beneficial side effects in that it has identified areas where more data needs to be collected at the scene of the crime and also identified important gaps in the knowledge of criminal behaviour. In addition it has been found useful as a training tool in that once it is established how the expert does his job, this knowledge can be used to train more human beings and aid further development of expert systems.

Commercial conclusions

Expert systems have been introduced, particularly by sales executives, as THE answer to all problems. An expert system is not capable of solving every problem, it must be used within defined domains. No program listing or data in a database can capture the infinite complexity of the world. Additionally, as the world is changing, any successful model of today's situation is likely to be invalidated tomorrow. There are two reasons why it will be impossible to develop a system which contains every last up-to-date detail of expert knowledge;

a) in some domains, new knowledge is discovered every day. In these circumstances continually updating the system would be impossible

b) in some domains, new knowledge is developed as a result of a 'domino effect'

Liebowitz (1987) provides a list of common fallacies about expert systems.

Do not forget that an expert system is just a piece of software and will suffer, as with any software system, from such problems as bugs, mains spikes, response time problems, crashes and human interference. Computer programs cannot avoid human error as they are designed, built and used by humans. Through the widespread use of a program containing a 'mistake', that 'mistake' can be rapidly propagated and amplified.

An expert system should not take a decision, it should display the consequences of various courses of action, provide any other available relevant information and let the human user make the final decision.

Beynon-Davies (1988) noted that a high proportion of commercial data processing could benefit from the application of expert system technology. Successful expert systems will be those which easily integrate into existing practice, particularly as there is no sharp boundary between conventional and expert system applications.

Expert systems are being more widely applied to commercial and industrial problems. It is fortunate for education that these tools, developed for commercial purposes, can also be applied to a variety of educational applications. Part Three of this thesis investigates these potential and actual applications.

PART THREE

EDUCATIONAL APPLICATIONS

THE EDUCATIONAL USE OF COMPUTERS

It is the intention in this chapter to place the current and future educational applications of computer technology into context and also to look at some of the issues associated with innovation.

The history of the use of computers in education has already been adequately chronicled elsewhere. See for example O'Shea and Self (1983), although Chorover (1984) argued that

> "only time and experience will tell whether or not the computerisation of education will actually revolutionise the ways in which we teach and learn"

AI and education share concerns about the nature of, and how to communicate, expertise. Although AI research is comparatively new, there is a very old and basic question at the root of the use of AI in education. It is 'what are the aims and objectives of education ?'

The primary question to be posed when considering the use of computers in the classroom is whether their presence and use will improve the learning situation. This was noted by Ellis (1984)

> "Thinking about the computer's role in education does not mean thinking about computers. It means thinking about education."

This suggests that educational policy makers should look beyond the technical aspects and consider not what the computer can do, but what the learner can do with the computer. The relationship between the student and teacher is an example of one of the social, rather than purely educational, changes which will result from the introduction of computers. Many American schools and colleges are adopting policies which increase the numbers of computers in use. Rogers (1984) and Bray (1984) detail some examples, with the latter providing particular emphasis on Clarkson University. Carnegie-Mellon University has developed arguably the most computer-intensive campus in the world. This development, with particular emphasis on the social implications, is chronicled by Kiesler and Sproull (1987).

Turkle (1984) develops a concept of placing human users at the centre of any analysis of computer use and on the purposes of that use, rather than the traditional concentration on the technology. Lieberman (1986) follows this line by suggesting that one way in which education can benefit from AI is that it can lead to putting more powerful computers into the hands of less sophisticated users, arguing that the more 'intelligent' the machine becomes, the easier it can be to program. Note though, the distinction between 'using' and 'programming' a computer. In systems intended for beginners, ease of programming may be the primary criterion, although the spread of microcomputers through education brought with it the myth that you need to enable pupils to learn 'how' to program. This was criticised by Aleksander (1984)

> "teaching people to make current computer structures and program them when the research community is endeavouring to alter such structures out of recognition .... seems sheer lunacy."

The myth could be considered to be a version of a more enlightened view, as described by Papert (1980) and Lawler (1984), where the emphasis would be to learn 'through' programming. This issue reappears in chapter twenty seven where it is argued that an educational application of KBS technology is to learn through the process of creating and/or using expert systems.

Traditional CAL programs are incapable of solving the problems which they set or the capability for solving the problem is limited to that method programmed into the system or by using algorithms which do not help the teaching/learning process. Dreyfuss and Dreyfuss (1986) accept that there is a place for computers within education, but consider, as does Self (1974, 1985), that most present day software is inappropriate.

With hindsight, it has become apparent that the use of many of the early CAL programs achieved little. Nevertheless this, perhaps, could be seen as a part of the 'learning curve' which education had to endure. It is also apparent that it is a long learning curve and education does not have a particularly good history of establishing innovations that involve long learning curves.

Curricular applications will fall into two broad groups;

a) those systems that are essentially teaching systems (eg Sleeman and Brown 1982)

b) those that are teaching aids in the

sense that they facilitate learning by doing (eg Papert 1980)
Yazdani (1986) provides arguments to show that neither are yet capable of being used in education, but that in the future these different components are likely to merge.
Barker (1987) identified two broad types of educational expert systems; advisory and instructional.

There would appear to be several areas throughout education where expert system technology could have a role to play. These possible functions could be as:

a) research tools (eg in teacher training Wood 1986a)

b) decision support and planning (eg school management)

c) curriculum resources

d) simulations

e) the core of a tutoring system

f) reservoirs of knowledge

g) a means of exploring knowledge

h) as a source of 'new' knowledge through induction

i) a means of expressing educational theories

There are two users of an expert system;

a) the 'expert' who puts the knowledge in. This may not be just a one-way process because the computer cannot jump to conclusions, relying on precise logical reasoning. Hence, 'experts' are forced to evaluate their own reasoning and the 'experts' may gain insight or greater understanding of certain domain features by going through the knowledge elicitation process.

b) and the student who makes use of that knowledge.

It is also important to note in an educational context, that the builder of the knowledge base does not have to be an 'expert' because some learning will take place during the research for, and the construction of, the knowledge base.

Advanced technology offers an extension of distance learning, moving education out of the classroom and into the home and place of work. The provision of open learning systems (OLS), particularly in Further and Higher Education is a trend that has grown during this decade. OLS can take various forms as described in CNAA (1981). Allan (1984) considers the theoretical requirements for, and the practical problems involved in, the development of a computerised information retrieval system that could aid OLS. The role of the expert system in these circumstances could be that of an intermediary or 'manager' to aid in the access of resources. These resources could be in traditional library form or as an OLS database. An expert system could also take on a 'teaching' role. The creation of such systems has potential, but the practical problems are seen as being immense. Cowan (1986) identified the 'seven deadly sins', weaknesses in OLS, which reduce the effectiveness of such systems. These are not technical matters, but fundamental curriculum considerations. The technology offers much, but may also offer Cowan's seventh sin of "an impersonal approach".

One of the underlying premises of 'self-organised learning' (Thomas and Harri-Augstein 1985, Todd 1988) is that many people never learn how to learn. The self-organised learner acts as a researcher within the particular domain. This is not an easy task for the learner but one which needs to be supported and it may be that a developing role for expert systems within education is that of 'learning advisors' or 'learning managers'.

In the last decade education has come under increasing scrutiny with political demands for greater efficiency and productivity. Purely economic comparisons between schools and factories cannot be legitimately made. However this is not to deny the need for a continued reappraisal of the aims of education. Computer technology has provided the vehicle for productivity improvements in many economic sectors. These improvements have been achieved largely by automating manufacturing, or other processes. It must be questioned if automating the teacher's role is possible or indeed desirable. The automation of large sections of industry has also had various social consequences. Education is a social activity and although computers offer potential for improvements, the social nature cannot, or should not, be overlooked. It is impossible to quantify all aspects of the educational process.

If society's future needs include the creation of a well-educated and flexible workforce, with learning in school followed by re-educating and training throughout adulthood becoming the norm, this would need the development of flexible learning systems, which take account of a variety of learning styles. In the planning and delivery of courses, tutors need to identify appropriate teaching and learning strategies. The style can range from the traditional didactic lecturing to almost complete student autonomy where the learning can be self-directed, self-paced and even self-accessed. The individualised, self-paced, mode of learning through the use of CAL has much to commend it in educational terms. The addition of an 'intelligent' module increases the strength of the argument. This is a utopian view, the reality may be some distance away, but it should not stop us trying.

Educational innovation

The relationship between research, development and innovation in education remains a fundamental problem. Why some educational innovations have survived, while others have disappeared without trace, may depend upon the actual innovation, the context of the innovation, the presence or absence of management skills of the innovator, a combination of these, or possibly some other factors. Computer technology in general is primarily not an educational innovation, but rather an educational application of a technological innovation. Ruthven (1985) noted that

"We should have learned four things from the experience of the sixties;

(1) that educational innovation is about aims and values as well as methods

(2) that an educational technology is not just a set of tools, but a way of using those tools

(3) there is no such thing as a 'teacher-proof' or 'pupil-proof' package or technology

(4) that innovation which ignores the experiences, practices, expectations and values which teachers and pupils bring with them, will either fail to establish itself, or be assimilated to those value systems"

Curriculum innovation is difficult as there are many factors which militate against change and change for the sake of change is probably worse than no action at all. Any curriculum development must be experimental and if new materials cannot commend themselves, on their merits, to teachers, they do not deserve to make any headway. Stenhouse (1975) made the point that curriculum development should start from a problem and work to a solution rather than working from the solution. This matches the need for commercial applications to be 'needs-led' as reported by Baker (1984) and Turner (1985) and discussed in Part Two. Implicit in curriculum innovation are a change in values concerning, for example, what pupils should learn and how they learn it, about subject matter content and about new ways of looking at the curriculum. As noted by, for example, Bartram et al (1986)

> "The way in which a system operates and the ways in which users interact with that system should be internally consistent and wherever possible consistent with other systems and with population stereotypes or expectations."

Hence a successful innovation must succeed at changing established values and overcoming this inertia is far from a foregone conclusion. Any educational innovation requires initial positive intervention and commitment, a point echoed by Ennals (1987) and Watson (1987). The development of expert systems and their potential educational applications provides, arguably, one of the most threatening curriculum innovations, as it strikes at the heart of many basic educational principles. An expert system is a program which encapsulates the knowledge of an expert in a particular domain, and can be used to provide advice and answer questions and also to provide an explanation of the logic by which the conclusions have been reached. This covers the central part of the role of a teacher and expert system developments are of relevance to those working in education.

Any innovation is threatening to the quiet status quo of the lives of a teacher and Bryant (1979) provides a further discussion of the issues of the psychology of resistance to change.

The curriculum issues associated with the introduction of expert system technology into education were highlighted by Piddock (1987). Technical considerations must be less important than curriculum development issues. There may be, in the long term, changes to almost all of the characteristics of modern education, either as a direct result of the educational use of the technology or indirectly through changes to society.

This background of rapid change must be viewed in the light of the effects upon society in general and education in particular. That is, apart from the inability of education to keep pace (it has been suggested that the educational half-life is at least one generation), but also the need to produce broad-based individuals capable of understanding and communicating over multiple fields. AI may have produced a problem for education, but, as will be discussed in chapter thirty two, it may well produce the solution.

Innovation, both in education and commerce, will only take place if Foggo's formula (Paine 1986) applies. This formula, although without mathematical precision, appears to have validity.

$$C = f\ (X, Y, Z,) > I$$

C = change or innovation
X = the perceived need for change
Y = there is a clear goal
Z = the first step to take is known
I = the investment (cash and/or human)

An analysis of the factors which determine the uptake of any commercial innovation is complex. The device must be cost effective and perceived as useful, change will only occur when the factors of (f) are seen to be greater than 'I'. Paine also noted that

> "factors are not absolute criteria, but relative criteria. If the investment is perceived to be too great, change will not occur."

This matches the 'organisational health' concept of Miles (1965), in that it does not matter how visible an innovation is, it will not be adopted unless the adopting unit is on the lookout for the innovation and is prepared to experiment with the innovation in their specific setting. This is as true of commercial applications as it is of education. Cotterell et al (1988) noted that

> "Success in education does not depend upon the production of complete, perfect automatic systems to run complex industrial processes, nor do the software systems used have to be complex."

Although education does not share the same objectives as commerce and industry, expert systems offer an innovation worthy of experimentation within the educational setting. In the next chapter attention turns to the application of AI in education.

AI IN EDUCATION

Research into Artificial Intelligence is a comparatively recent area of study and the use of AI within education is an even more recent innovation, beginning with the use of LOGO, if it is recognised that LOGO is an AI language. This final point is not universally accepted, but it is not a discussion which I wish to pursue.

The creation of LOGO 'microworlds' or 'toolkits', as outlined by Sharples and Finlayson (1985), has the advantage that the complexities of the procedures of the language can be hidden from the user. This is an alternative use of LOGO to that described by Papert (1980), Goodyear (1984) and Harvey (1984). They have claimed that the use of LOGO can develop logical thinking and transferable skills through writing programs. The user learns through the building of simple LOGO procedures and gradually increasing the complexity. Lawler (1985) noted that

> "a significant part of the educational power of 'learning through programming' rests in students' freedom to experiment with programs and to engage in inquisitive, speculative tinkering"

From my own experience, a close study of children learning through LOGO shows that the role of the teacher is critical. The teacher doesn't teach explicitly, but there is a need for a very sophisticated level of guidance. This latter point was emphasised by Brown and Burton (1982)

> "knowing when to intervene is a difficult decision, too much intervention can hinder learning as well as support it."

For further discussions of the use and applications of LOGO, see, for example, Papert (1980), Ross (1983) and Allen (1984).

Of the two major AI languages, LISP has not found a great deal of use within UK schools. However, the use of PROLOG within the educational environment has been the subject of a number of investigations. Ennals (1984) provides an account of the early classroom applications of Prolog. There are inherent features of Prolog which Wild (1987) describes and uses to commend its use in the field of teaching and learning. Ennals (1983) and Kowalski (1984) present the case for adopting the development of logical thinking in children as the starting point for the consideration of the use of computers in schools. Stern (1987), however, notes that learning specific skills in logic may be useful, but is less important than developing a method of learning through the building of knowledge bases.

Thorne (1986) identified three approaches to the application of expert systems within the classroom;

a) learning and teaching about expert systems and how they work

b) an expert system constructed by the teacher and available for the pupils to use

c) developed as extensions of the use of database packages and such programs as 'The Tree of Knowledge' (Acornsoft), where the pupils use shells and toolkits to build their own expert systems

I believe that it is in the construction of knowledge bases that the major potential lies, as noted by Kemp et al (1988)

> "expert systems are generally much more interesting to write than to use"

The exercise of constructing a knowledge base allows learners to clarify their understanding and may also promote discussion of topics where difficulties may lie, with both teachers and fellow students.

The concept of placing students in the role of knowledge engineers is not only simple, but powerful. It is the activity of eliciting, acquiring and representing knowledge with the aim of producing an expert system which holds the power. As was noted in Part Two, in the commercial world, there are difficulties in constructing usable systems, but it is the process of constructing the system rather than the final product which is of greater educational interest. As with LOGO, above, and as noted by Stern (1987)

> "one learns by developing and modelling one's own conceptual structures in an interactive reconstructable medium"

As a consequence of the educational emphasis being placed on the process rather than the product, there are potential applications across the curriculum and across a variety of school age groups. Casey (1986) provides one of a number of reports in the domain of chemistry. Weinberg et al (1987) report on ORESS (Oxidation and Reduction Expert System Shell), written in Prolog, and Bateman (1987) describes a system that has been constructed using APES and Prolog. In the latter case, through the use of a menu-driven interface, the pupil can select chemical problems, attempt solutions or ask for appropriate information or explanations. Biology provided the domain for the work reported by Rasmussen (1987) and Geyer (1988). Evaluative trials using ADEX (Briggs 1987) in biology and geography were reported by Hassell (1987) and Webb (1987). Both these reports indicated that the most valuable application could be in the construction of knowledge bases of just a few rules, particularly if the building exercise promoted discussion about the specific domain. This is of particular interest when comparing the commercial and educational applications of expert systems. As was reported in chapter fourteen, d'Agapeyeff (1984,1987), referring to commercial applications, stated that small systems can be useful and this will be of increasing importance as systems are now easier to build. Hassell and Webb also demonstrated that an expert system shell can provide a suitable tool for qualitative modelling.

This finding was further reinforced in the development of Q-Vitamins (Christian-Carter 1987). The first system was written using MITSI (Briggs 1984) to take advantage of the memory available on the RML Nimbus. However, the knowledge base, with some queries, produced duplicated and lengthy information which could only be analysed after it had been dumped to a printer. A new system, 'Q', was therefore written to provide easier access to the knowledge base. 'Q' provides a framework to generate an expert system out of a MITSI knowledge base by manipulating the MITSI rules somewhat differently. This is an important aspect of knowledge-based computing in that rules from one system can often be removed and entered into another system that will then manipulate them differently.

Computer modelling is not restricted to the sciences. Briggs et al (1984), Goble (1987) and O'Connell (1987) report on applications in the humanities. Goble provided details of work with sixth form geography students using

Prolog to produce a range of computer models. O'Connell reported on the use of Prolog-based toolkits (LINX and DETECT) with secondary pupils and noted that

> "pupils are able to develop a range of sophisticated programs which represent their developing understanding and knowledge of a problem"

In the area of history, Nicol et al (1986) observed that

> "pupils' historical understanding develops through their processing of historical sources .... a major element in developing historical understanding is the application of logical reasoning to a discrete mass of data."

The 'toolkits' approach is also applicable to the Primary curriculum, as described by Watson (1987). However it was noted that "the introduction of these toolkits in the projects has been facilitated by my presence (and knowledge). They will not succeed in schools generally unless accompanied by extended in-service education."

A support structure will be needed if further toolkits and systems are to be developed and disseminated. Nevertheless, as was considered in chapter twenty four, this comment would be equally applicable to many educational innovations.

The use of Prolog in teaching foreign languages was reported by Barchan et al (1985) and Yazdani (1987). The thrust of their work was in the production of an 'intelligent' teaching system rather than in learning through the building and use of expert systems.

Raffan (1987) and Sibbett (1987) report on the use of toolkits (SLOTS) with dyslexic pupils, noting that

> "a major triumph of SLOTS is that it has succeeded in catching and maintaining the child's attention throughout the session"

This 'success' may be a function of using the computer just as much as using SLOTS, although their project has permitted the exploration of using a knowledge structuring tool in the classroom. The results with dyslexic children have been sufficiently encouraging to extend the project to develop the data handling facilities.

Claridge and Nicol (1986) provide an appraisal of the use of Xi in a classroom situation. Some of their criticisms (eg unsuitable documentation and tutorial) refer specifically to Xi. In fairness to Expertech (the producers of Xi), the system was not specifically designed for this use, but Bainbridge (1986) and Bignold (1986) noted

that demonstrating commercial software to sixth form computer science students was of considerable benefit by itself. The improved version (Xi Plus) may have answered many of the earlier criticisms. Their other comments refer to the educational application of Xi, noting that an 'education' version of the software may be an improvement.
Following on from the conclusion (Briggs 1987) that simple shells, not 'cut-down' commercial systems would be required if educational staff and students were to be allowed to explore some of the potential uses of expert systems, Briggs developed EGSPERT. This was designed as a system with a simple syntax and query system which would be easy to use. To evaluate the system, Briggs worked with a number of members of staff within Further Education and from a variety of disciplines and varying degrees of computer awareness. As a result of this work, a number of other systems were developed as part of an 'Expert Systems Starter Pack' (Briggs 1987).

ADEX, one of the systems in the Pack, is an advisory system which has a similar knowledge representation language to EGSPERT. This meant that it could be quickly learned and a number of systems were developed covering a wide range of curriculum areas. Some of these examples were also included in the Starter Pack. The step-by-step explanation provided by ADEX can become tedious and perhaps a graphical trace through the tree diagram of the rule base would prove to be a useful addition to the explanation facility. It is interesting to note that few current commercial systems have attempted to integrate graphics with text. As the technology develops it is expected that links with graphics and animation will become more common.

ES/P Advisor was used by a class who had previously used LOGO, a factor which was seen as important, the most positive conclusion reached by Treadwell (1986) was

> "the children surprised us in terms of their adaptability and the use of problem solving strategies in

overcoming the difficulties encountered."

Expert Ease is an example of a system which induces both the rules and the questions (both based upon information supplied by the user). Walton (1986) noted that the system would allow an attribute (a question) to be deleted and would then induce a new rule set. This allowed the pupils to experiment with the inclusion or exclusion of various factors and assisted them in their choice of what data to collect and how to structure it.

There are some analogues for curriculum applications in the types of problems addressed by the commercial applications discussed in chapter fourteen. For example (Briggs 1987, Cotterell et al 1988)

a) choosing a product - hairdressing preparation, site for industrial or commercial development

b) giving advice - health education, road safety, which specific modules of a modular course to study next

c) diagnosis - central heating system, the performance of the local football team

d) explaining a process - chemical, legal, electoral or psychological

e) analysing data - census data, trade directories

The purpose of using the system in the classroom may be to act as a catalyst and stimulate discussion. It was reported in Cotterell et al (1988) that the use of a system dealing with social class provoked a heated debate. McCarthy (1986) reported that

> "the language involved in using a knowledge base is of great importance
> .... the discussion during the creation of a knowledge base is extremely important and the children extended their vocabulary, understanding and general communication skills."

Outside the classroom

There are also examples of the use of expert systems within education, but not specifically in the classroom. The ESTE Project (Expert Systems in Teacher Education) at Sussex University is one such example. This collaborative project, in the field of teacher education, is also concerned with the implications for applications within the general social sciences area. Wood (1986a) notes that our knowledge and understanding of how and why things happen in social situations, such as the classroom, is far less precise than in scientific areas. Additionally it was noted that theories of social situations tend to lack the predictive power normally associated with scientific theories. This lack of prescriptive power effectively precludes making advice prescriptive and the Sussex team have adopted a similar critiquing stance to Miller (1984). The preliminary knowledge acquisition technique of the project was to invite comments from experienced teachers on video recordings of trainee teachers in the classroom. This allowed the identification of seven factors;

control	motivation
learning	forgetting
comprehension	communication
relationship	

which were used as a basis for modelling classroom practice.

Ennals and Cotterell (1985) describe a program, developed as part of FEU research project 141 ('Computer-based educational consultancy'), which categorises educational objectives, checks the course against requirements and recommendations and selects teaching strategies. This program was designed to aid teachers, but a similar program could be developed to aid students. If the learner was working on his own, the system could suggest routes through the work and advise upon learning strategies. Other possibilities included advice on what courses to take and the course requirements needed and future career guidance. CET have investigated the role of computerised guidance systems within the Manpower Services Commission's (now called the Training Commission) Training Access Points (TAP) project (Humphries 1986). Newton (1988) noted that just the provision of information by itself would be insufficient and most adults would benefit from counselling and guidance. Logica are developing a system, as part of the TAP initiative, to help users analyse for themselves what training requirements they might have and provide the basis for a business plan for discussion with a counsellor.

An example of a working system is APE, developed at Israel's Bar Llan University. However, APE doesn't incorporate a 'simulation' of the human judgement taken by the University Registrar who remains the final arbiter of whether a student is eligible for a degree.

Advisory systems could be developed in a wide variety of other education areas. For example;

a) a careers advisor (providing advice on the qualifications required for entry to various careers)

b) a student grant aid advisor (providing advice on the level and types of aid available to students)

c) a Health and Safety at school advisor (an expert reference manual for these regulations, incorporating both checking and recommendation of good practice)

d) a Governor's advisor (providing advice on relevant procedures and legislation)

A potential application of expert systems, in the light of the current concern with monitoring school performance, would be as a performance indicator analyst. Discussions have already taken place between Coopers and Lybrand and the D.E.S. and, if implemented, this could be a parallel development to the PIA system developed by Coopers and Lybrand for use by the Regional Health Authorities.

The present proposals to increase the number of assessment tasks that schools are required to administer may provide another potential role for expert systems. This application could be in the area of data interpretation and analysis and may be closely linked to the 'PIA' system above.

Staff development will be a central issue as the new technology becomes increasingly used and the organisation and management of Colleges may change, but, as Ennals and Cotterell (1985) note, simply putting an 'intelligent workstation' on a College principal's desk will not necessarily improve the management of that College. This highlights the distinction between expert systems to be used by non-experts and 'expert' workstations to be used by experts. In the former case, the machine solves a problem in the same manner as a human expert, in the latter, the machine provides a set of tools for the human expert to use to solve a problem.

As was demonstrated in Part Two, expert systems have been applied to a variety of commercial tasks. One product, the NCC Expert System Starter Pack developed by the Alvey Project, was aimed at increasing awareness of the technology. This product was developed primarily for the attention of commercial organisations. Nevertheless some Further and Higher Education Colleges purchased the Pack and the following chapter reports on a survey of their subsequent use of the Pack.

THE USE OF THE NCC EXPERT SYSTEMS STARTER PACK IN FURTHER AND HIGHER EDUCATION

The Alvey Programme (Alvey 1982) recognised the need to bring to the attention of a wide spectrum of UK organisations the potential importance of IKBS techniques in general and expert systems in particular. Hence the IKBS programme of the Alvey Project was divided into four sub-programmes, namely

IKBS demonstrators
Research themes, Projects and Clubs
Support Infrastructure
IKBS Awareness

One of the initiatives aimed at increasing this awareness, was to commission the National Computing Centre to produce and sell a practical introductory Pack to IKBS. The Pack, called the Alvey/NCC Expert Systems Starter Pack contained complete reference documentation, four training guides and a specially written introduction to the basic concepts of the technology. It also contained four demonstrator versions of commercially available expert system software packages which were chosen to demonstrate various techniques as exemplars of current technology. They were;

a) Expert-Ease (Intelligent Terminals Ltd) was chosen to demonstrate the technique of rule induction and the principle of forward reasoning.

b) Micro Expert (Intelligent Systems International.) demonstrates the treatment of uncertain data and illustrates backward reasoning.

c) ES/P Advisor (Expert Systems International) demonstrates a technique known as 'text animation', which is particularly useful for providing information based on written text.

d) Micro Synics, the final piece of software in the Pack, is a dialogue generator, not an expert system shell, but was included to demonstrate the important ability to adapt systems to the end-user.

It was stressed when the Pack was launched, in May 1985, that it was intended purely as an introductory and training product and that it was not intended for actual development work.

This chapter reports on a survey that I carried out in association with NCC into the use of the Pack within Further and Higher Education Institutions. It was carried out with two aims in mind;

a) to assess the impact of the Pack on the educational establishments that had purchased it. (Over 30% of the original sales of the Starter Pack had gone to educational establishments)

b) to establish the present, and possible future, scope and direction of the use of expert systems within the educational environment.

At the end of August 1986, 123 questionnaires were sent out to educational establishments that had purchased the Alvey/NCC Expert Systems Starter Pack. Forty replies, representing 36 organisations, were received. These were analysed as phase 1 of the survey. This was followed, in August 1987, by a second questionnaire sent to those respondents of the first questionnaire. This second questionnaire (phase 2) sought to establish the continuing use of the Pack and the pattern of development. Fourteen replies were received in response to this second questionnaire.

In a further attempt to monitor the development of the use of the Pack, an additional twenty sets of the first questionnaire were sent to educational organisations that had purchased the Pack during the period July 1986 - July 1987. However, only three replies were received in response to this initiative.

It will be noted, therefore, that the sample was not a random selection of educational establishments concerned with the use of expert systems. Hence the results must be viewed in the context of a self-selected group. Nevertheless, I believe that the data does provide valuable information about the use of expert systems within the educational environment.

The multiple choice sections of the questionnaire were organised on the 4-point 'PAGE' scale (Poor, Adequate, Good, Excellent). No detailed statistical analysis has been undertaken and all percentages have been rounded, as the purpose of the exercise was not to provide a survey detailed to the final decimal point, but to indicate the general scope of work and possible future trends.

The figures in the angular brackets <> refer to sections of the questionnaire, details of which can be found in Appendix 6.

Phase 1 Questionnaire responses

Details of respondents

41 replies were actually received, but 1 of

them was duplicated so the final sample was 40, making the response rate for the survey 32%. They were treated as individual entries even though they may have come from the same organisation. Indeed it was noticeable that in these cases, departments from the same organisation had differing needs and consequently made different responses to the questions.

The sample obtained was representative of the full database of NCC Starter Pack educational customers as shown by the following table.

	All customers	Respondents
Universities	40%	50%
Polytechnics	24%	25%
H.E. Colleges	9%	7%
Others	27%	18%

Furthermore the sample was geographically representative England 26, Scotland 5, Wales 3, Northern Ireland 1 and Eire 1.

There was no correlation between membership of NCC and purchase of the Pack, only 25% were members of NCC, indicating that purchase of the Pack was not merely to support their NCC membership.

The responses were received from a variety of departments, but the majority came from Computing (55%) and Engineering (20%). Among the other departments were Metallurgy, Psychology, Business and Management, Statistics and Agriculture. This indicates the diversity of interest in the technology and its application. Only 1 reply was received from a specialist AI/Expert Systems Department.

<u>Section 1</u>

Questions relating specifically to the Alvey/NCC Expert Systems Starter Pack

<u>The Pack itself - software <1.2></u>

The vast majority of the respondents (80%) had purchased the IBM version (which is again representative of the full database). There were complaints about the inability to install or run particular packages on particular machine configurations. The other major criticism that was mentioned frequently was the limitation of not being able to SAVE examples when using one of the packages supplied (Expert Ease).

There was considerable divergence of opinion relating to the software in the Pack. This perhaps shows up the preconceptions and expectations that people had of the Pack. It also, undoubtedly, relates to the applications that people may have considered, or planned, for each piece of software. This reflects the different styles of the software in the Pack.

Micro Expert

Only six organisations had not used the software and this was seen as easy to use by those who used it.

Ease of use 'adequate' (50%) 'good' (40%)

Relevance 'adequate' (28%) 'good' (59%)

Micro Synics

47% of the organisations replied that they had not used the software, but of those that had used it, this piece of software produced the most extreme responses ranging from "totally useless" to "real value for money". The relevance of the product was questioned and generally it was felt that it was a 'make weight' in the Pack. However the specific use of the package was appreciated by those groups that had made good use of it.

Expert Ease

Six organisations had not used it, but, as the name suggests, it was seen as being particularly easy to use by those that had used it as shown by the following responses.

Ease of use 'adequate' (17%) 'good' (47%) 'excellent' (27%)
Relevance 'adequate' (24%) 'good' (55%).

ES/P Advisor

This was probably the best received piece of software in the Pack. Only 4 organisations had not used the software and of those that had, it was well rated.

Ease of use 'adequate' (25%) 'good' (57%)
Relevance 'adequate' (20%) 'good' (67%).

The Pack itself - documentation

The documentation was well received, apart from the single criticism

> "there were too many different items of documentation - difficult to find out which one was relevant".

This was countered by complimentary remarks about the training guides. It perhaps shows that you cannot please all of the people... and there may well have been just as much criticism if the documentation had been packaged as a single item.

Ease of use 'adequate' (31%) 'good' (53%)
Relevance 'adequate' (17%) 'good' (72%)

The complete Pack

Although there were specific criticisms of particular items in the Pack, the general feeling was that the Pack, as a whole, was of use and of reasonable value.

Pack usefulness 'adequate' (35%) 'good' (39%)
Pack value 'adequate' (47%) 'good' (31%)

Nevertheless, it must be remembered that these were responses from organisations that had already purchased the Pack. As this was a postal questionnaire of actual Pack customers, it was impossible to produce a control group against whom to compare and contrast responses. However, informal discussions with some educational institutions who were interested in expert systems, but who had not purchased the Pack, indicated a variety of responses for not purchasing the Pack. These included;

a) the departments were already sufficiently aware of, or skilled in the use of, the technology

b) the departments already possessed appropriate shells or languages

c) the technology was not seen as being appropriate to the work of the department at this time

d) the department was unaware of either the product or the technology

e) lack of finance

The use of the Pack

Number and frequency of use (1.3 & 1.4)

Some organisations had purchased the Pack as a central resource and no formal records of its use had been kept. However, from details supplied by the other respondents, it is clear that the Pack has been put to a variety of uses. The use of the Pack does vary from place to place, but a picture, albeit painted with broad sweeping strokes, of the typical use of the Pack can be obtained. It is a single member of the teaching staff working on a project with a small number of students on a weekly basis. This picture, though, does hide the examples of a single student using the Pack intensively or a large group using the Pack intermittently over a long period of time.

Actual format of use <1.5>

Note that 'introduction' and 'hands-on experience' do not figure in the central column as they were not available responses on the questionnaire. Although 'hands-on experience' is not specifically seen as having a major role to play in the future, there will be a significant amount of 'hands-on experience' by virtue of the use of the Pack as courseware and as a demonstration, familiarisation, awareness tool.

<u>Uses of the Pack</u>

	Reason for buying Pack	Use of the Pack	Future use of Pack
Awareness	8 %	18 %	13 %
Familiarisation	10 %	18 %	15 %
Demonstrations	12 %	16 %	11 %
Evaluation	23 %	14 %	6 %
Teaching	15 %	10 %	10 %
Courseware	0 %	9 %	10 %
Research	2 %	7 %	5 %
Seminars	0 %	4 %	4 %
Prototyping	2 %	2 %	0 %
Development	0 %	2 %	2 %
Introduction	23 %	- %	4 %
Hands-on exp.	8 %	- %	4 %

The 'evaluation' and 'introduction' roles had fallen off considerably showing the speed of change of the available software. It is interesting to note that no organisation purchased the Pack to use as a piece of courseware, but that 9% mentioned it in terms of a current use and 10% as a future use.

Planned use of the Pack

(Aug '86 - Aug '87) <1.6>

In the same way that the Pack is being used for a variety of jobs at present, its projected use will also be very variable ranging from "gathering dust" on the one hand, to "incorporation into teaching" and "intensive use on courses" on the other. This does show the discrepancy in needs or interests between the various institutions. Some of the respondents are working at the 'leading edge' (though not specifically with the Pack) while others have the job of introducing and teaching about expert systems to a wide variety of students.

While some places had no specific plans for the Pack, many institutions planned to continue to use it in the role of a demonstration, awareness and familiarisation tool.

'Prototyping' was not mentioned as a specific future use of the Pack, probably because the limitations of the software, as regards 'real development work', particularly the inability to SAVE Expert Ease files, had been realised. It was noted that institutions were planning to purchase, or had already purchased, other products for this development work.

Other comments <1.7>

Many of the replies in this section concerned installation and running problems. The ideal, but unattainable system, would be a suite of software that ran without complaint on any hardware configuration !

It was noted by several institutions that there was the need to update the Pack, perhaps including a frame-based shell or a languages Pack. This point was also noted by NCC in the planning and development of their Expert Systems Resource Pack (NCC 1987).

The Pack was seen, by some observers, as falling between two stools, in that

> "it was too sophisticated to be used independently by naive users but contained too little technical detail for those wishing to understand more about the operation of the software."

The problem of the most appropriate level of detail to include in the Pack was something which NCC debated for some time. One of the reasons for the production of the Resource Pack (NCC 1987) was to provide a more current sample of software and to supply more detail for those users who required it.

Many of the comments made about the Pack reminded me of similar comments made during two telephone surveys that I conducted with a sample of all the Pack customers (January and August 1986). The conclusion that I draw from this is that the majority of purchasers of the Pack, whether educational institutions or commercial organisations, were, at the time of purchase, in the same position, namely that of novices, or alternatively, they may have had to cater for 'novices'. The speed and direction of development that has taken place since then is very variable. It would appear to be a function of the specific nature of the organisation and its particular needs, perceived or actual. Although it concerned both groups, generally speaking the educational institutions were less concerned about producing a working system.

I did look for specific criteria upon which the various institutions claimed that aspects of the Pack were not satisfactory, but I didn't find any conclusion that could be stated with any certainty, except as is stated elsewhere in this chapter, the difference between customer expectation and Pack performance. This varied considerably from item to item and from discipline to discipline.

Nevertheless, the Pack has helped to generate considerable interest in expert systems and there would certainly appear to be 'educational' potential in this technology.

Phase 2 Questionnaire Responses

There were 14 responses to this questionnaire (again a response rate of 33%). These respondents were representative of the first sample.

As regards the use of the Pack, the replies ranged from 'none' to 'fairly extensive', but the overall finding was that it was being used less frequently and by fewer people. The reasons for this could be summarised as the software was dated and had limitations and also change of personnel. This latter factor is particularly significant on two counts. The first survey found that in many cases there was a single member of staff who was pioneering, or attempting to pioneer, the development of expert systems within the organisation. It also highlights the limited number of 'educational expert system practitioners'.

There was not a significant change in the balance of the users.

Users	1986	1987
Teachers	41%	38%
Research staff	21%	21%
Postgraduates)	33%	24%
Undergraduates)		17%

The following table compares the actual and future uses of the Pack as expressed by the 1986 survey with the actual use as expressed by the 1987 survey

	1986 actual%	1986 future%	1987 actual%
demonstrations	16	11	14
seminars	4	4	4
teaching	10	10	11
courseware	9	10	9
awareness	18	13	18
evaluation	14	6	9
familiarisation	18	15	20
research	7	5	7
prototyping	2	0	2
development	2	2	7

This shows that the Pack does still have a continuing use, even though some of the respondents saw the software as being dated. This continuing use is particularly as a teaching tool and especially in an awareness and familiarisation role. This is very much the role that the Pack was originally designed to serve. The fact that none of the initial sample organisations purchased the Pack to use as a piece of courseware, but that 9% mentioned it (phase 1) in terms of a current use and 10% as a future use supports the fact of its continuing use. As the above table shows this estimate of its use in this role has held up well.

The fact that its use in evaluation has not fallen off as far as was predicted by the phase 1 survey also indicates that although the software in the Pack may have limitations and appear, in some quarters, to be dated, it is still a valuable tool to use on 'basic' courses with beginners. Those users who have moved on to 'better' products, including the NCC Expert Systems Resource Pack (NCC 1987), are probably working on 'non-basic' activities and may have specific 'advanced' uses.

The NCC Expert Systems Resource Pack was launched in Spring 1987 as the follow-up product to the Starter Pack. It contains versions of five currently available expert system software, each one including demonstration applications and some also including on-line tutorials. The five tools are;

Crystal (Intelligent Environments)

Expert Edge (Helix)

Savoir (ISI Ltd)

Xi Plus (Expertech)

Super Expert (Intelligent Terminals)

In addition to full documentation for the above software, also included in the Resource Pack are a booklet written by the Treasury's Central Computer and Telecommunications Agency on the principles involved in undertaking an expert system project, a booklet on knowledge acquisition, in-depth details of 26 case studies of UK applications and a software directory database supplied on two discs.

Although the Starter Pack itself is perhaps not being as widely used, the respondents are using other software products on a range of activities from MSc courses to IT courses and from research to actual development work. The phase 1 survey respondents mentioned Lisp (12%), Prolog (30%) and Poplog (3%) and 23 other products. This compares with Lisp (14%), Prolog (21%) and Poplog (12%) and 16 other software products that were mentioned by the phase 2 respondents. The shift in the Prolog and Poplog figures may be of interest, but the smaller number of other products mentioned is not seen as significant bearing in mind the reduced number of respondents.

The survey also sought to establish the interest in 'intelligent' teaching systems (ICAI). The finding is that there is a general trend of increasing awareness and interest in both expert systems and ICAI. A point of semantic interest here, is the distinction between 'having an interest in' something and actually doing something about it. Although there is an increase in terms of actual numbers, there are still only a small number of actual applications. The following are possible reasons for the lack of developments in this area;

Insufficient hardware) Insufficient software)	Money
More pressing demands) Inappropriate to the) current work of the) department)	Time
Unsuitable hardware) Unsuitable software)	Technological development
Lack of appropriate) personnel)	Manpower

A further reason, particularly relating to developments in ICAI, is that it remains a complex application. Chapter twenty eight contains a discussion of this issue in further detail.

Survey conclusions

It is clear that, despite the various criticisms, the Pack has played a significant and valuable role in increasing interest and awareness in this technology. It has introduced the ideas to a large number of people from a wide range of disciplines and there is now a wide spectrum of actual and potential educational developments.

Certainly the survey has thrown up a number of varied potential roles for such systems within the educational environment, for example

a) Intelligent front-end to software packages

b) Intelligent help system

c) Teacher support (relief from 'mundane' tasks)

d) Student's assistant

e) Advisor system for conceptual problems

f) Cognitive modelling tool

g) Distance learning

h) Links with interactive video

i) Self instruction systems (particularly with procedurally-based techniques)

j) Another teaching aid/method for the pedagogue

Ennals and Cotterell (1985) and Briggs (1986) provide examples of the use of expert systems in Further Education within various curriculum areas. They clearly see the use of such systems as tools which pervade the teaching of each subject area. 'New' subject areas may develop as technological advances facilitate a more individual approach to learning by the student and a more knowledge based approach by the teacher. In the case of the latter, this will involve greater use of library facilities where information would be stored in a variety of forms (videodisc, floppy disc, microfiche as well as on paper). The nature of teaching/learning will change even more when the stage is reached when libraries, as described above, are linked together via a communications network.

What the educational world needs, as a minimum, is an awareness of developments in expert systems and better still the opportunities to experiment with the technology. The recent allocation of over £3 million of Manpower Services Commission money to fund AI applications in education and training over the next three years is a welcome, if limited, move in this direction. However, the fulfilled and unfulfilled potential of expert systems in education is such that it cannot be ignored, but money alone is not the answer. The potential is accompanied, as noted by Piddock (1987), by a whole range of curriculum issues which are noted in the following chapter.

AI AND TRAINING

In this chapter, I note the distinction between 'education', which I see as a general concept, and 'training', which I see as being more specific and a sub-set of education. Harmon and King (1985) provide a three-way categorisation;

a) education - which achieves changes in performance by providing conceptual principles that allow a person to think in abstract terms

b) job-aid - which could take the form of a checklist, handbook, calculator etc which allows the user to come up with the correct answer but without necessarily knowing a great deal about the subject

c) training - a middle course where some theoretical information is provided but in the context of carrying out a particular procedure or accomplishing a specific goal

This could be linked to the statement from chapter two that we use experts in a mix of explainers (a), problem solvers (b) and information providers (c).

Additionally I note the distinction between CAL (the term typically used within the school situation) and CBT (the term usually applied within industrial training). Training is more specific and didactic than education. In education students acquire knowledge before they know exactly how they will use it. Training covers a very broad area ranging from CBT to advice giving systems. CBT is designed to teach and test with a view to making a person more 'expert' in some area of expertise. Advice giving systems are designed to be 'expert' themselves and through their use a person can become more 'expert' by seeing the 'correct' questions to ask and the consequent conclusions given.

CBT has been available for a couple of decades but much of it has a poor student model and in many cases merely acts as an inflexible 'page turner'. Conventional CBT presents the material in a predetermined way

and fails to adapt to the specific needs of each student. As expert systems separate the knowledge and the way in which it is used, it may provide the opportunity for the coursewriter to concentrate upon the subject material and the student to explore the knowledge in a more flexible way. Few people are able to combine the domain knowledge with the authoring skills necessary to turn the former into a training package.
Knowledge-based techniques, with their strengths in quick prototyping and easy modification, may begin to improve the power and convenience of the authoring process and the final outcome. A further advantage is that expert systems offer more flexible control over the branching through a piece of courseware because they are particularly adept at dealing with complex decisions.

Heaford (1983) noted, as one of the limitations of conventional CBT, that

> "both users and writers soon run out of system space"

indicating that they reach the edges of the system, both in theoretical and practical terms, and it was questioned as to what happens at that stage.

CBT has become an invaluable tool in some of UK's largest companies. Laurillard (1986) found that 26% of UK companies were already users of CBT and 50% of the remainder were planning to introduce it in the near future. The figures for the USA use of CBT run at about twice the UK rate. Cost effectiveness is an important consideration in applying industrial CBT. Industrial training is usually aimed at teaching people to do a specific job and as it is done in the firm's time, there is a pressure for the training to be cost effective. The costs involved with interactive video (IV) can prove effective if it liberates training staff from the classroom or reduces the time employees take off work to be trained. This is more likely to be a consideration in the commercial, rather than in the school, sector.

A commercial system must be put to work on a daily basis and demonstrate its cost effectiveness, although as was discussed in chapter twenty, the benefits were more often intangibles such as 'improved customer service' rather than direct financial savings. However there are large numbers of instances where it has been found that there is an additional bonus in the form of a training benefit through the use of expert systems. As examples consider;

HIFSQ, QUICK and SPHINX (medical systems)
GPSI and PROUST (concerned with Pascal programming)
SECOFOR (oil well drilling)

all of which are reported to have teaching applications over and above their specific applications. Further evidence of this additional training capability is provided by the case studies in NCC (1987)

> "it has also been a useful training aid" (ICI)

"the system has proved to be very useful as a training aid" (Courtaulds)

"the system results in an upskilling of technical staff" (British Gas)

"increases the skill and productivity" (Intelligent Applications)

In addition, the training must be effective within an acceptable timescale and with appropriate use of resources. This has restricted the use to limited domains and four major hurdles have prevented the CBT market from 'taking off'. They are;

a) the cost of developing good courseware

b) the inherent technical limitations

c) the lack of widespread delivery infrastructure

d) the wooden inability of conventional

CBT to adjust the kind of material that it presents to the students

The inability to place the learner at the centre is particularly important as it is a feature of human learning that the learner expects to be taught in an intelligent fashion. To watch a student plough through a rigid piece of courseware is sufficient evidence to appreciate this concept. AI offers the possibility of much greater flexibility and the ready availability of an expert consultant program can improve the training environment in industrial settings.

In everyday life, much learning takes place without explicit instruction and much instruction takes place which produces little successful learning. This could be likened to the 'watching Nellie' concept (Dixon 1988) as a low level form of commercial training. It was reported by Jones and Davies (1986) that intensive use of APRES has had the result that less highly qualified staff have been able to operate at an advanced level. Through such use, it was also found that they had absorbed much of the expertise encoded in the system and were increasingly able to work without reference to the system. The case studies in NCC (1987) provide three further examples;

> "Although envisaged as a planning tool, the system was found to be very effective in enabling new users to 'get up to speed'" (Thomas Cook)

> "The system can be used as a training aid not only for operators but also for new staff joining the team" (ICI)

> "The more junior officers will have 'learnt' much of the expertise of the best expert" (Universities Pensions)

Similarly Stone (1988) noted that

> "the use of expert systems would seem to lend itself to the experiential learning approach and represents a considerable advance on the 'knowledge first - questions after' implicit in much of the currently available computer software."

Durant (1987) noted that the Japanese Fifth Generation computer project emphasised educated and adaptable people as a key national resource. However it should be noted, as was reported by Boseley (1987), that the Japanese are also concerned about the perceived defects, such as high suicide rate and low output of personnel with imagination and initiative from their educational system. Ennals (1986a) believes that a similar versatility should be built into the British workforce.

> "We should get away from the idea that the purpose of vocational

education and training is to fill specific manpower slots."

Coopers and Lybrand (1985) noted that

> "Faced with international competition and an ever increasing pace of technological change, the survival and prosperity of a developed nation depends on a highly trained and adaptable workforce. The UK has been slow to accept the importance of training which is regarded as an avoidable cost."

Certainly when commercial budgets are restricted, training budgets have been among the first to come under pressure. This as Dixon (1988) noted is

> "a serious, strategic mistake, when major economic technological restructuring is occurring"

With the coming population reduction in the 16-35 age group, education, training and re-training will become more and more important.

Given this low priority afforded to training, it was decided, in an evaluation project reported by Eary (1987), to disguise the training element (training by stealth) within another software package. The package, aimed at owner/managers of small firms (60> employees) in manufacturing and industrial services, provides a 'Position Audit' (developed by Durham University Business School) of the firm's current strengths and weaknesses. As the users work through the package they are are asked questions about their business. Combinations of answers lead to interim findings and inferences from a number of interim findings are reported as key findings as soon as they are derived. As each section is completed a conclusion of the firm's performance is reported and the user has the option of examining the key findings that contributed to the conclusion and in turn the interim findings that contributed to the key findings.

This approach was labelled 'training by stealth' with the training being provided through experience of the analysis produced by the software plus explanation facilities. Considerable effort was placed on providing suitable explanation facilities. Explanation-based learning is further described in chapter thirty.

An expert system can often serve as a useful example of a good strategy in approaching a problem, which might be helpful in a training context. However a novice may not be able to follow the reasoning steps of an expert because the expert's processes of organisation and compilation are, by definition, far advanced. A psychological assessment of this can be found in Anderson (1982).

Dixon (1986) sees AI technology and CBT merging in the long term to produce more 'intelligent' courseware. However, as Naughton (1986) noted

> "merely embedding expert systems in conventional CBT systems is not, in itself, sufficient to make significant enhancements to them."

Taking GUIDON (Shortliffe 1976) as an example, the system does work, but all it does by way of explanation is to provide a rule-trace. This is not the same as an explanation that would be provided by a human tutor. The rule-trace may be satisfactory if the user is an expert, but not if the user is a novice student.

There is also a benefit to be obtained by the 'expert', as noted in NCC (1987)

> ".... improved communication for training" (ICI)

> "building the system helped the expert when he came to

communicating his knowledge to others" (Water Research Centre)

Naughton (1986) suggests that AI research should be treated as a distinctive way of thinking about computers. Useful insights into thinking and training may be obtained by viewing from an AI perspective.

The technique of interactive video is an exciting prospect, even though the current cost is prohibitive. The BBC 'Domesday' video discs are one such example. Within the concept of interactivity lies one of the stumbling blocks to increased development in this field. Although there may be debate about various theories which purport to explain the underlying psychological phenomenon needed for learning to take place, few psychologists would disagree that learning is enhanced by the active participation of the student. In conventional learning systems, the student is invited to enter into a 'dialogue' with the machine, but the rigidity of the machine severely limits the quality of the dialogue. Only when the machine's capacity and capability to conduct an 'intelligent dialogue' has been engineered will we see the realisation of the central concept of interactivity itself.

One of the successes of the Alvey programme, as discussed in chapter 19, was the development of the IKBS Community Clubs (Appendix 3). The ITDU at Kingston College is developing a Training Club along the same lines. For example, Singer-Link Miles, who produce flight simulators, are involved with ITDU to explore the possibility of developing a flight simulator which could become the front-end of an expert system for training pilots. Hammond (1986) reports on a project at Hewlett Packard's Bristol Research Laboratory where they have developed a photolithography advisor which is a frame-based expert system linked to a video. This enables the system to demonstrate visually what the expert would do if he was there.

In a project designed to improve the effectiveness of part of the manufacturing operation, Birds Eye Walls (Gloucester) are developing a knowledge-based system capable of providing information in an interactive text and video form at the work place. The development of a higher level IV (Intelligent Video) by combining AI research techniques, such as those into machine intelligence, with IV technology offers significant enhancement to an already exciting potential.

This highlights the fact that tuition delivered at the time and place of need tends to stick. One of the problems of conventional training courses is that a proportion of the material learned on courses is lost between the course and the place of work. Dixon (1988) noted that

> "most people learn (the specifics of) most of what they do in their jobs through 'watching Nellie' i.e. watching from day one how somebody actually doing the job proceeds and

> copying them ....
> in the real world a lot can be, and has been, learnt from 'watching Nellie', even though Nellie is not a tutor and rarely gives instruction."

If it can be shown that expert systems can provide efficient and cost effective training at the place of work, then education may be 'released' from vocational considerations and could concentrate upon helping students to learn how to learn and be able to take advantage of vocational training and retraining. Developing the ability of 'learning how to learn' will increase in importance as it becomes more and more difficult to predict what specific knowledge and skills will be essential for the future.

In the following chapter, attention turns from systems which aid learning to the much more complex area of developing 'intelligent' systems which teach.

TEACHING AND INTELLIGENT TUTORING

The training applications of expert systems (discussed in the previous chapter) and their application in information retrieval systems (discussed in chapter twenty nine) provide areas of educational interest which could be termed 'teaching and learning aids'. In this chapter attention is focussed on the development of 'teaching systems' which is another area of interest.

It is worthy of note that Papert (1980) did not discuss the uses which people will find for computers, but rather the power of computer environments to affect the way in which people think and learn, stressing the educational value of the stages of programming, including the initial and continuing analysis, identifying problems and debugging.

The transfer of knowledge and expertise through teaching and training is a sophisticated and complex process and the idea that teachers will be totally replaced by computer tutors is misguided. The knowledge of the subject and the knowledge of teaching consists of more than merely knowing facts and rules. Dreyfuss and Dreyfuss (1986) suggested that

> "computers will not become first rate teachers unless researchers can solve four basic problems, the need to talk, to listen, to know and to coach"

It will be many years before this level of machine expertise is reached.

Conventional CAL techniques assume that all students are the same and the route through the course is predetermined by the teacher (teacher-driven software). In the real world, no two students learn in the same way and so the route through the course should depend upon the student (student-driven software).

In the use of CAL, there will come a point where the student passes to a level of work beyond that which the computer, either by virtue of the inherent algorithm or the limits of the domain knowledge, is able to present. Here the human teacher will be needed. It is interesting to contrast this with the 'intellectual' growth of expert systems which are dynamic, growing with use. During each interaction, the knowledge base is questioned, refined and enlarged. The result is that the expert system eventually develops expertise beyond that of a single expert.

Self (1974, 1985) was critical of the standard of much of the available CAL because of the limited student model inherent in the program. Even assuming that we could build into a system a more sophisticated model, one of the hardest aspects is for the model to track the student's advancing knowledge. Any system must include a means of evaluating the performance of the student and such procedures are independent of the lesson content. This need to develop a model capable of evaluating and adapting in response to the evaluation is not a trivial pursuit. Beynon (1985) noted the need to keep in mind the distinction between a discovery teaching strategy and a discovery learning model. The latter approach was proposed by Self (1985) in noting the limitations of using the expert system approach to designing CAL. He argued that the emphasis should be on the learner rather than on the domain knowledge. The 'domain-centred' approach does not concern itself with considerations of what the learner does and knows.

INTELLIGENT TUTORING SYSTEMS

There is a hierarchy of 'educational' computer systems ranging from those which present material, those which also incorporate a tutoring component and those which can actually demonstrate proficiency in the subject being taught. Intelligent tutoring systems (ITS) involve the application of AI techniques to the educational process, a concept echoed by Clancey (1984)

> "an ITS is a computer program that uses AI techniques for representing knowledge and carrying on an interaction with a student"

and Goodyear (1987)

> "ITS are computer-based systems which use techniques from AI to provide a dynamically adaptive learning environment for individual students"

An expert system is a system which contains knowledge and methods of manipulating and presenting that knowledge. French (1987), reporting on the use of TEST (Training Expert System Tool), defined intelligent tutoring systems as

> "an expert system that possesses methods specific to the field of

> training and may have information underpinning the use of the training methods."

However Briggs (1987) identified the fact that ITS are vastly more complex than expert systems due to the fact that ITS must be more active. The standard dialogue with an expert system consists of the system asking a lot of questions and this appears to be a weak method of helping students to learn. Whatever method of knowledge representation is used, a tutoring system needs to use that knowledge in a greater variety of ways than an expert system. In chapter twenty two it was reported that the knowledge contained in MYCIN was too narrow to be used to teach students to be primary diagnosticians and hence the development of NEOMYCIN and GUIDON. Furthermore if the formal representation of the knowledge is used in part of an explanation (as in Xi and APES), then the explanation will require interpreting in that context. Under such circumstances the interpretation may actually hinder learning.

A number of commercial systems were listed in chapter twenty seven as providing training applications in addition to their specific applications. However the relevant point here is that these systems were not tutoring systems but expert systems developed for the commercial purposes as discussed in chapter fifteen.

ITS can be viewed as an evolutionary development, the earliest stages of which were the linear programs based on the 'operant conditioning' theories of Skinner (1958). Crowder (1959) identified the need for a branching structure and the premise that the program should take account of the actions of the student

> "the essential problem is that of controlling a communication process by the use of feedback"

(Appendix 5 provides diagrams of four early CBT algorithms).

AI training programs generate the method to be used in solving a problem by calling upon the range of reasoning methods given to it by the programmer whereas the conventional program merely implements the reasoning already done by the programmer.

Traditionally the subject matter is organised into frames which are tied together with a branching strategy. AI-based software allows a greater range of learner responses. One intermediate aim may be to incorporate AI techniques into authoring systems to facilitate more complex branching options.

Smallwood (1970) gave an indication of the complexity of developing teaching systems by calculating that a system with only five instructional alternatives at each branch and two possible student responses at each instructional alternative would need to consider ten billion possible student routes merely to cover the next ten presentations to that student.

The CAL programs that have been developed since those early days have improved, partly as a result of improved software engineering, but the inherent pedagogic sophistication has not kept pace with the technological development. The algorithms used in CAL programs mean that they can vary their response, to a limited degree, with the student's level of understanding. Through the use of branching programs with predefined 'choice-points' it is possible to offer the students the 'twin gods of CAL', as proposed by O'Shea and Self (1983)

> "richer feedback and a much greater degree of individualised learning"

However, Yazdani (1984) considered that

> "the behaviourist theory of learning still shines through, they are all basically learning by being told"

One of the aims of the work of Kimball (1982) and O'Shea (1982) was the development of systems which would themselves develop an improved teaching strategy by using the student responses as a basis for the decision making process. In the selection of advice to the student, Kimball's system used the method applied by the student, rather than the actual answer.

Although there is research effort (eg Leinhardt and Greeno 1986) into developing a theory of tutoring expertise, Self (1987b) noted that

> "There is no well developed 'theory of tutoring' which can justify any particular use of student models in a particular context. For example even if an ICAI system is correct in identifying a student misconception, it is difficult to provide a theoretical reason for selecting a particular remedial action."

In coaching systems, there is a clear distinction, as identified by Sleeman and Brown (1982), between the type of diagnostic task (why did the student not make a better attempt ?) and diagnosing why the student actually made the error. The difference lies in the fact that when a student makes an actual mistake, the system has the opportunity to continue the tutorial from that point. However, if the error was one of omission, then the system would not be in a position to know if the omission was because the student did not know the specific item or knew it but decided not to apply it or didn't know how to apply it. Furthermore, the error may be caused by straightforward misunderstanding or be the result of indirect environmental factors such as carelessness or tiredness.

Burton and Brown (1979) noted that there are two major, but related issues, which must be borne in mind when developing such a system. They are deciding when to intervene and deciding what to do when intervention has taken place. A tutor, human or machine, may have the capacity to provide the solution to a problem, but providing that solution at too early a stage, may deprive the student of a learning experience. Similarly, providing the solution at too late a stage, or providing too much help, may frustrate the student. Unless the solution is provided at an appropriate time, learning will be incomplete. If it was a difficult task to decide when to intervene, deciding how to offer help provides a bigger challenge to the system. Unless the help is provided in an appropriate form, learning again will be incomplete. Note that these are pedagogical, not technological, issues which are further discussed by Burton and Brown (1979).

Potential advances in AI research were the basis of Suppes (1979) who expressed the hope that

> "we should expect by 1990 CAI courses of considerable pedagogical

> and psychological sophistication. The student should expect penetrating and sophisticated things to be said to him about the character of his work"

It is questionable if this level of sophistication will be achieved by 1990. Nevertheless, this feedback is important because results presented by Atkinson (1976) indicated that the learner is not a particularly effective decision maker in guiding the learning process. In any dialogue, there are difficulties associated with variation of emphasis in expression and wording. AI-based systems may offer a richer dialogue than the severely limited dialogue which takes place in conventional CAL. There is a need to appreciate the distinction between 'what' to say and deciding 'how' to say it. This is not implemented in the simpler approaches to the generation of textual responses (eg MYCIN translates data structures).

Several systems are based on using errors or 'bugs' as the starting point for constructing the next step for the learner. This hypothesis, gaining support from the work of Sussman (1975), has as the central theme, the idea that errors may highlight areas of basic weakness or highlight the stage at which the extrapolation or processing technique adopted by the student broke down.

Donaldson (1963) identified three classes of error;

a) structural (failure to appreciate some essential principle or relationship)

b) arbitrary (inconsistency in applying a technique)

c) executive (errors that occur during the execution of a problem)

The work of Donaldson was followed by Matz (1982), working in the field of algebra, who proposed that errors are the result of attempting to extrapolate from existing knowledge to a new situation. This led to the development of three similar categories of common errors;

a) poor knowledge, where the basic concepts have not been successfully mastered

b) 'extrapolation' errors where the basic concepts are sound, but the 'transfer' or extrapolation was unsuccessful

c) processing errors

A major problem with the 'bugs' paradigm is that they must apply within a very narrow domain. A look at the skill lattice for subtraction provided by Burton (1982) will give some indication of the complexity of even a relatively narrow area of mathematics.

Goldstein (1979) noted that maintaining a simplistic viewpoint of the teaching/learning process allowed researchers to concentrate upon fundamental issues such as knowledge representation. However such a stance doesn't tackle 'real' problems unless the viewpoint is expanded or developed.

"expert-based CAI allows only for

> the definition of 'simulated students' formed from subsets of the expert's skills"

Goldstein continued, and defended the use of 'simulated students'

> "Simulated students do allow the testing of systems and may yield an insight for the human teacher observing their performance"

There are a number of existing systems which demonstrate the flexibility of learning programs based on AI. O'Shea (1981) and Sleeman and Brown (1982) highlighted a number of significant systems which aimed to be diagnostic in nature rather than drill and practice. DEBUGGY is a system which is claimed (Burton 1982) to perform as well as human teachers in diagnosing problems in subtraction. However, the complexity of the task performed by the system increases dramatically when the domain is widened. Nevertheless, Attisha and Yazdani (1983,

1984) have attempted to extend the ideas of DEBUGGY to cover addition and multiplication. These systems contain much domain knowledge which may not be as readily available in other subjects. The computer, as teacher, would need to establish, from the responses of the pupil, the particular response that the machine should make, hence the system would 'learn' about the pupil. The best human teachers never stop learning about their subject and their pupils. Machine learning, which is discussed in chapter thirty, may offer a possible solution here.

The 'bug' model involves an exhaustive identification and cataloguing of all the various types of error that the student can make within a given domain. Success along this avenue, therefore, requires detailed analysis of the task or problem. The system then identifies the particular 'bug' from the catalogue and this controls the next stage of the process. The major problem with this approach is that it is virtually impossible to catalogue every possible error that a student could make and furthermore if the student lacks or misuses a particular concept, this may be due to the concept being too difficult for the student's current level of expertise or due to a lack of information. The 'buggy' model of learning assumes that the student has a 'bug-ridden' knowledge of the domain and it is the task of the teacher to identify and help the student to correct these 'bugs'.

A limitation of any student model associated with a specific tutoring strategy is that any strategy will only match a given student model. It is for research to evaluate how the learning processes differ from domain to domain. A variety of information relating to personal details, background and cognitive factors and the experiences of the student all have a bearing upon the way in which that particular student will learn. To incorporate multiple student models and multiple teaching strategies is a complex task and will provide one focus for much further research.

Another approach is to employ the 'expert' model, which involves noting where, and to what extent, the student differs from the 'expert' model contained within the system. There are problems with this approach particularly where there is a major difference between student and 'expert'. Furthermore the system must have a dynamic student model to adapt to the fact that the student will change over time and the actual changes may be dependent upon whether successful learning had taken place.

One of the early assumptions made by research workers on ITS was that if you can completely model a student's problem solving behaviour on a range of tasks, then remedial treatment is straightforward. This is a weak assumption because even if a 'complete model' was attainable, it does not recognise that the provision and organisation of remedial treatment is a complex business. In particular the recognition of previously unencountered behaviour patterns is a very open-ended problem. This may be acceptable within limited domains, as with much CBT, but rapidly becomes unmanageable when applied to less specific domains where the potential for error is much greater.

The distinction between ITS and CBT is largely a matter of degree in that the latter directs the student towards the correct answer whereas the basic premise of the former is that students learn more effectively by discovering answers for themselves. However as noted by Wenger (1986)

> "ITS cannot be regarded as an extension of traditional CAI/CBT. Making the move to explicit representation of various forms of knowledge and reasoning within the system changes the enterprise. An ITS cannot be seen as a presentation device for material prepared externally to the system."

In ITS research, great emphasis is placed on understanding student misconceptions. Briggs

(1988) noted that

> "training someone to troubleshoot and fix a compact disc player presents very different teaching problems to helping someone understand the background and effect of the rise of multi-national companies. Building an ITS to help teach one may prove easier than the other."

Burton and Brown (1979) and Goldstein (1979) advocate that the philosophy behind the coaching strategy should be on the basis that hints should be provided at increasing levels of specificity and determined from the program's best next step.

The important features of MYCIN (Shortliffe 1976), as they relate to the development of ICAL were identified by O'Shea and Self (1983) as;

a) an appropriate way of representing

knowledge (production rules)

b) a reasonable initial set of facts and relations which can easily be extended

c) a natural and comprehensible mode of reasoning

d) a dialogue capability

e) an ability to explain its decision making processes

The use of an expert system as a teacher would involve the system incorporating into the knowledge base, a variety of forms of knowledge which in total could involve thousands and possibly millions of rules. Hartley (1973) identified the following five components;

a) student model

b) student history

c) teaching administration

d) teaching generator

e) teaching strategy

The following categorisation of the multi-faceted forms of knowledge that would be required by such systems is based on Lantz et al (1983) and Allan (1984);

a) subject knowledge - knowledge of the subject matter and problem solving in the area to be taught

b) teacher knowledge - knowledge of pupils, experiences and expectations, role perceptions, legal restraints

c) educational knowledge - learning theory, teaching, assessment

d) system knowledge - an awareness and appreciation of the variety of uses and approaches to the system

e) 'setting' knowledge - concerning the learning environment, national and local education structure, organisation and objectives, educational climate

This compares with the commonly agreed components of most ICAI systems, as identified by Dede (1986); a knowledge base, student and pedagogical models and a user interface. This categorisation follows Meighan (1981) who identified these theories as playing a role in the process of education;

a) a theory of knowledge

b) a theory of learning and the learner's role

c) a theory of teaching and the teacher's role

d) additional theories

The subject knowledge is possibly the easiest to quantify and codify. Certain types of knowledge have been demonstrated to be quantifiable (eg the medical knowledge contained within MYCIN Shortliffe 1976). However, the subject knowledge by itself is useless unless your view of education follows the 'Hydraulic Theory of Education' (Davies 1969) and is merely the transfer of knowledge. If the latter is the case, then computer technology offers little, the knowledge could easily be transferred on tablets of stone.

Self (1985) has pointed out that there is a contradiction associated with the development of systems which contain the expertise to perform, or teach, a particular task. If the development was successful then it would render this area of human endeavour redundant and this would further reduce, or even eliminate, the need to teach this particular skill.

It has been shown that intelligent tutoring systems offer an effective system within a narrow domain, as do expert systems within limited commercial domains, as was discussed earlier. Sharples (1984) noted that

> "after a decade of research on ICAI, the resulting systems are highly domain-specific: to extend any program beyond its narrow subject area would require it to be largely rewritten."

There are some general qualities, over and above domain-specific qualities, that must be possessed by a 'good' ITS. For example efficiency (being able to respond to the student in a short time) and robustness (being able to cope with a wide variety of unexpected student responses).

For more general learning by 'problem solving', Papert (1980) and Lawler (1984) have argued for the development of 'microworlds'. A central theme of this thesis is that 'ideas' learned in one domain can be generalised into a variety of domains. A further development is the AI programming environment, four examples of which are presented by Yazdani (1984).

```
environments        >
                     >     teaching  >
                      >    aids       >
intelligent          >                >
database systems    >                 >
                                       >
                                        >   AI
                                       >
intelligent         >                 >
tutoring systems     >                >
                      >    teaching   >
                     >     systems   >
ICAL                >
```

The 'teaching aid' category matches the

commercial 'job-aid' categorisation of Harmon and King (1985) mentioned in chapter twenty seven.

These approaches are like twigs on a branch each with their respective strengths and weaknesses. The focus of environment-based learning is on the activity of the learner, an idea based on Piaget (1971). The strength of an ITS is that it will contain both an explicit theory of learning and domain expertise. It is easier to evaluate the performance of an ITS because it will, by nature, be well defined for a given curriculum. The strength of the open-ended exploratory learning environment is also a weakness because it is difficult to establish their impact and effectiveness and the environment would need to be placed in context by a human teacher.

A straightforward use of an expert system as a tutor would be for the knowledge base to contain knowledge of a variety of examples within a particular domain. The system could generate descriptions of situations or problems which the student would have to analyse or solve. If the analysis offered by the student differed from that of the system, the system could criticise and explain the reasons for its solution, allowing the student to identify the possible error. ATTENDING (Miller 1984) is an example of such a critiquing program.

A discussion of the differences between traditional CAL and AI-based programs, with particular emphasis on MENO-II, is found in Soloway et al (1983). A further discussion of the potential of AI in CAL can be found in Goldstein (1979). Howe (1987) suggests that building a dynamic model of the user's knowledge is perhaps the key issue in the area of ICAL. O'Shea and Self (1983) point out that many of the fundamental difficulties in building intelligent computer tutors lie within the field of psychology. Boden (1977) suggested that

"the development of automatic

tutors will go hand in hand with increasing psychological appreciation of the way in which humans build and progressively modify internal representations of concepts and skills."

Programs that have been developed have been largely constructed to investigate a small subset of the larger problem. For example, SCHOLAR (Carbonell 1970) was developed to investigate tutorial dialogue, WHY (Stevens and Collins 1978) to investigate student modelling and diagnose conceptual errors, the LISP tutor, GREATERP (Anderson and Jeffries 1985) is psychologically based on Anderson's ACT* theory and Goldstein intended to use THE GENETIC GRAPH (Goldstein 1979) to explore and analyse Piagetian learning.

Hence as many of the ITS projects are involved in basic academic research, they may assist in understanding how knowledge is assimilated or why explanations may be misinterpreted. This remains a long term goal and is unlikely to directly affect classroom practice for many years.

An area where there is likely to be a more immediate application is the use of expert systems as aids in the process of retrieving information from large and complex databases. The following chapter looks at this application which, as a result of the 'information explosion', is likely to become of increasing importance.

INTELLIGENT INFORMATION RETRIEVAL SYSTEMS

As discussed in chapter seven, knowledge, at its simplest, is structured information and it is that very structure that enables machines to process knowledge. The techniques of knowledge representation, vital in the structuring process, were discussed in chapter nine. The links with commercial data processing were considered in chapter nineteen.

Database systems might be seen as a subset of knowledge-based systems, the main difference being that the database systems can only regurgitate the information and do not allow rule-based inference. The following diagram based on Turner (1988) indicates the increasing level of sophistication. Similarly, Addis (1985) identified a number of systems as enhancements of data retrieval systems.

```
data storage       ) DBMS  )       )            )
data retrieval     )       )       )            )
-------------------------) M     )            )
data structuring         ) I     )            )
data distribution        ) S     )   D        )
report generation        )       )   S        )  K
exception reporting      )       )   S        )  B
---------------------------------)            )  D
mathematical modelling           )            )  S
algorithmic processing           )            )  S
----------------------------------------------)
rules of thumb                                )
uncertainty                                   )
judgements                                    )
explanations                                  )
=============================================
creativity        )         the human
intuition         )         features which
common sense      )         machines cannot
emotion           )         provide
=============================================
```

If students are to be helped in handling a vast amount of detailed and specific information then an 'intelligent' knowledge base would appear to have much to offer. Evans (1986) identifies the advantages of using an expert system as an 'intelligent database'

> "since it is able to elicit exactly what you want, but related to the composition of its own data structure
>
> ..... the addition of new knowledge

is not a matter of integrating new data into existing files by going through a 'create' package; it is done by the system itself ..... so that the new material is structured in relation to the existing format, cross-referenced with it and absorbed into the whole."

The task of retrieving the data from a database provides a level of difficulty, particularly for novices. Query optimisers are a means by which relational database management systems determine the best way to execute file maintenance and retrieval operations. This is a difficult issue even for the modern range of relational database management systems and it will become more complex when considering distributed databases where data is distributed over a number of sites and machines. Under these circumstances the number of query processing possibilities multiplies geometrically and expert systems may offer a means of improving the efficiency of retrieving data from a variety of database systems. BD11 is an example offering intelligent retrieval from a distributed database.

Data which is held in a database may be easy to extract. What is not so easy to extract is the information that has not been explicitly stored, but which can be deduced from the stored information. It is certain that the storage of information will be revolutionised by knowledge bases. Cooper (1984) and Defude (1984) discuss the relationship of AI to information retrieval (IR), concluding that sophisticated systems requiring the accurate logical deduction of information from a knowledge base will become possible only if an adequate interdisciplinary theory of language and logic is developed and that the expert system architecture will depend upon whether it is a general or specialised IR system that is under consideration.

The following systems, detailed in Appendix 4, come from a variety of domains and provide examples where an expert system assists in the process of knowledge retrieval;

CANSE	KBMS	PUBLI
COALS	KIWI	SMART
EPX	KNOES	WELDS
FIBER	LRS	
GPLAN	MIRIA	

Hafner (1981) considers intelligent search techniques within the domain of law and Zarri (1984) provides details of the inference operations and the principles underlying the architecture of the RESEDA system, another example of an 'intelligent' IR system.

As on-line information systems grow, both in size and number, there will be an increasing need for the system to provide help to guide the user through the system. This could take the form of a guided tour (an electronic version of the tourist guide) or a filter (highlighting important information and masking irrelevant information). By using AI techniques, this 'help' facility could be given a degree of 'intelligence' enabling the system to adjust the 'tour' or the 'filter' to take account of the specific user and his history in using the system. There would be enormous educational potential in the use of such 'electronic encyclopaedias'. IIDA is an example of a system which aids the searching of on-line scientific and bibliographic information retrieval systems. Scanlon and O'Shea (1987) note that there is a clear distinction between an encyclopedia as a medium for the storage and retrieval of knowledge which provides the information, but little in the way of tutoring and explanation and an encyclopedia as a tutor. The production of the latter is a massive challenge that has yet to be faced.

On-line databases may contain vast amounts of information, but just because the information is there doesn't mean that all users are able to easily access the information. It was reported by Fisher (1985) that 40% of the functionality of on-line help systems was either unused or unrealised. Using an expert system as an intelligent front end to an on-line database should provide the required information more efficiently (quicker, cheaper) and could turn out to be an important use of the technology in the future. This is an area which is being explored by an Alvey project (Alvey Mailshot 1987).

Any attempt to teach based simply on transmitting all knowledge will fail because of the vast and increasing quantity of knowledge. On pragmatic grounds alone, only a sub-set of 'total knowledge' can be taught, but through the teaching of selected knowledge, various approaches and methods can be introduced. The increasing shift in emphasis from 'what do you know ?' to 'do you know how to find out ?' will mean that the ability to organise information will be vital and it will become increasingly important for work with knowledge-based systems to have a high priority. Simply reading information from books provides a restricted opportunity for learning and Michie (1980) observed that there is a probable benefit for information to be 'pre-digested'. For example the artificial librarian's assistant would offer an 'intelligent' means of interrogating the library catalogue. Such a tool could have considerable benefit for research workers in many fields. Simon (1972) expressed the view that

> "the change in information processing techniques demands a fundamental change in the meaning attached to the familiar verb 'to know'
> .... 'to know' meant to have stored in one's memory in a way that facilitates recall when appropriate .... the whole emphasis in 'knowing' shifts from the storage or the actual possession of information to the process of using or having access to it."

This matched the much earlier view of Dr Johnson (18th April 1775)

> "knowledge is of two kinds, we know a subject ourselves, or we know

where we can find information upon it"

As we progress to an increasingly information-based society education will need to place more emphasis on the 'access to information' aspects rather than on the more traditional 'learning the information'. As was noted in chapter eighteen, decision making is dependent upon access to information.

The following section is based on my own experience of using several database programs (QUEST, GRASS, KEY) in the classroom and a variety of expert system shells. The use of database systems in the classroom is to collect and enter information, process that information in a variety of ways and extract information in a particular format to suit the specific application. Therefore I am dividing their use into three stages; data entry, data processing and interrogation.

Prior to entering the actual data, the data collection exercise itself can be a very valuable educational experience. Any knowledge based system (database or expert system) structures the information. Many reference books provide unstructured information (the Observers series being a notable exception) and the collection of information from a variety of such sources and structuring the information themselves was educationally valid. This structuring of information is a prerequisite for learning and those pupils who failed to collect their information in a suitably structured form, had problems entering the data into the database system.

An advantage of using a program like Xi is that the entry does not have to be sequential. Rules and data can be entered in any order and the system will be able to sort it out. As was discussed in chapter ten, 'new' knowledge may be provided by a number of systems which have been developed to 'learn' by induction.

The data processing aspect of the operation is usually of little interest to the end user, providing that the system works.

In my experience, the query language required for data interrogation by QUEST provided pupils with far more problems than were necessary. This was improved with menu-driven systems such as KEY and GRASS and the similar system employed by Xi which were much more user-friendly.

Children in school collect vast quantities of data, often just for the sake of it. The creation of a database allows the exploration of that information. If that database has been created using PROLOG, as described by Nicol and Dean (1984) then there is the additional benefit that the user has to specify 'what' is to be done, but does not have to specify 'how' to perform the specified task. SLOTS (Sibbett 1987) is a Prolog-based general purpose toolkit, produced at Imperial College, London, which enables children to create their own database, edit the data and query the knowledge base.

The SPIRAL (Schools Prolog Information Retrieval and Learning) Project at the University of Leicester (Sept 1985 - Aug 1987) investigated the use of micro-Prolog as a computer language to use in the writing of data handling packages. The project concluded that children need to understand the nature of information before they can be expected to use information retrieval programs and they can learn through experience of dealing with unstructured information about the desirability of structuring information. This can give opportunities to use information skills, but will not necessarily develop them. Additionally it was reported that considerable amounts of time need to be allowed so that children have adequate opportunity to discuss their work with others.

Access to, and the handling of, information is gaining in importance and it will continue to increase in tandem with the 'information explosion'. Expert system technology can assist in both these operations and their use in the area of information retrieval can provide a variety of learning situations. It is the complex concept of learning to which attention now turns.

LEARNING AND MACHINE LEARNING

In this chapter the fundamental educational concept of learning is considered. The commercial applications of expert systems were discussed in Part Two, but they were developed for commercial purposes and not for the the purpose of helping people to learn in the educational sense of the word. However this is not to say that commercial purposes and educational purposes are not compatible. It was reported in chapter twenty seven that it has been found that there is a learning element that accompanies the commercial use of expert systems. The educational applications that have been discussed in the previous chapters could enhance and extend the existing curriculum and provide the opportunity for new learning experiences.

What is learning ?

Learning is a difficult concept to define because of the complex and inter-related aspects of human behaviour. However, as was noted in chapter four, the ability to learn often features as one of the defining characteristics of intelligence.

Education has been a subject of study for many years and even now, we do not fully understand how learning takes place even though there are a vast number of educational 'experts'. For example, Norman (1985) is struck by how little is known about cognition and notes that it is a difficult matter to comprehend because there is more to learning than the accumulation of knowledge. Indeed a key feature of the nature of learning is its complexity.

Learning can be seen as a general term used to describe the process by which people (and machines) improve their performance perhaps by an increase in their knowledge or an improvement in their skills. Kolb (1984) suggested that learning is a continuous transformational process, emphasising the 'process' nature and observed that

> "experiental learning is a process that links education, work and personal development."

Learning is not a single activity, but it can be represented as a series of activities in a continuum, ranging from 'teacher-centred' activities (eg behaviour modelling) to 'learner-centred' activities (eg experiental learning).

Schank (Durham 1986) suggests that there may be a general principle that runs through a range of intelligent mental activities, theorising that we build up a series of 'explanation patterns' and then defining learning as the acquisition of these patterns. Understanding can then be viewed as the use of them with creativity seen as the process of modifying old patterns to fit a new reality.

There are two complementary reasons that are relevant here for studying learning;

a) to understand the process itself so that we may improve our understanding of what knowledge is and how it develops

b) through a greater understanding of knowledge and learning in humans to potentially build computers with the ability to learn

Barr and Feigenbaum (1982) stress that the most important factor affecting the design of any learning system is the quality and level of the information provided. For example information is of little use if it is presented in an inarticulate or unreliable form and this is equally true of expert systems. Four fundamental learning situations have been defined by Barr and Feigenbaum (1982).

a) rote - this is purely a process of memorisation and the only problem is the retrieval, upon demand, of the knowledge. It is not a particularly sophisticated mechanism, but it does form the basis of learning, in that all learning systems must remember previously learned knowledge so that it can be retrieved and applied in the future to new learning situations. Barr and Feigenbaum (1982) considered rote learning to be the worst way to learn as the material is largely dissociated from reality and as such is often meaningless. It could be argued that conventional computer systems work on a 'rote' basis in that they merely store and retrieve series of instructions.

b) learning by being told - every time that a tutor (human or machine) tells the student something, it is denying the student the opportunity for self discovery. Human tutors, often because of time pressures, interrupt and prevent the development of important cognitive skills especially the skill of detecting and using their own errors. The tutor must be able to recognise when the learner has reached a plateau or a problem and provide appropriate responses. Hence the problem is not only how to respond but also when, or indeed if, to intervene.

c) learning from example - if the example information provided is too specific or detailed, the learner must hypothesise, or induce, more general rules. Eary (1987) noted that an explanation of a question by using suitable examples is a technique that is recognised as being a particularly effective way of offering help. A partial solution to the problem of providing explanation in expert systems (discussed later in this chapter) may come through the provision of examples.

d) learning from analogy - the information provided is only relevant if it is concerned with an analogous performance and so the learner must discover the analogy and hypothesise analogous rules for its present performance task. In the classroom, this can work well if the teacher presents good examples in good pedagogical order. Learning is more difficult if the examples are inadequate, difficult to analogise or if presented in an unordered fashion.

Learning is evolutionary (in Darwinian terms) in that it proceeds forward but also includes 'jumps'. Children's learning does not follow a path of a position of 'truth' to a successive position of 'truth'. Their natural learning paths include false positions (shown by Piaget as a necessary part of the process of learning to think). Sometimes learning is primarily a process of accepting new information and integrating it into existing structures, at other times it is a matter of reorganising the information that is already present. Psychological research on concept learning (Wason and Johnson Laird 1968) confirms that people become confused if too many differences are presented at once. Mental activity has been pictured as a continuous structure building process (Chalmers, Crawley and Rose 1971). At the lowest level the child selects from unlearned responses or previously learned habits, as his experience grows, habits that are not helpful drop out and are replaced by the establishment of helpful habits. After solving many problems of a certain kind, a more organised pattern of responses that meet the demands of the situation is developed. Eventually the individual may organise simple learning sets into more complex patterns of learning sets, which in turn are available for transfer as units to a new situation and thus the child learns to cope with increasingly difficult patterns.

These learning sets have been identified as 'plans' and it was found that some 'plans' were memorised (eg learning the alphabet or counting to 10). However it was noted that as the numbers increase, then it is likely that the child works in terms of a set of rules or formulae for generating numbers. Mills, Galanter and Pribram (1960) reported that if a 'plan' is in frequent use, then it is more likely to be memorised . Compare this situation with a computer which could use a look-up table or a formula to calculate the answer to a numeric problem. A formula, being of a generalised nature, could always be used, whereas the look-up table, being specific in nature, may not be able to provide an answer for all situations.

In some cases, as identified by Boden (1977), learning is a process of accepting new information and integrating it into existing structures, whereas at other times it is a matter of reorganising the information that is already stored.

If a new task meshes well with what has been previously learned, then the earlier learning can be transferred with profit to the new situation, if not then the task is harder. Hence the most efficient method of teaching involves linking old to new, highlighting the relationship of the new problem to things that the child has already learned, rather than requiring the child to master more new information. Children at first solve problems by trial and error, only gradually does such behaviour give way to immediate solutions. Gagne (1965) developed a hierarchy of learning levels of which the 'shot in the dark' trial and error approach was seen as the lowest level and problem solving was viewed as the highest level of learning.

Psychological research into learning by, among others, Hilgard and Bowe (1966) and Bolles (1979) provides two conclusions which command universal assent. Short term memory (STM) which is limited to seven chunks is distinct from long term memory (LTM) which, for practical purposes is unlimited, and feedback must be provided quickly and is crucial to the acquisition of new skills. The importance of this immediacy of reinforcement is particularly relevant to learning new tasks and, as discussed in chapter twenty four, CAL, in general, does offer this facility.

Learning improves and extends existing knowledge. Winston, reported by Banks (1986), identified Martins's Law

> "You can only learn about what you already almost know"

In humans, the learning of any task involves the laying down, in the brain, of a new network of interconnected neurons. Some knowledge must exist within any learning system to enable that system to understand the information provided with the new problem, to enable it to hypothesise and to test and refine those hypotheses. Various modes of learning have been distinguished, the new knowledge may be assimilated into, or accommodated by, the existing knowledge (Piaget 1971) or it may require a major reorganisation to represent the knowledge more efficiently (Rumelhart and Norman 1983).

Can expert systems help human learning ?

Most specialists admit to having gaps in their knowledge and the process of building an expert system could help to identify such areas. Bainbridge (1981) found that examining one's thinking analytically, as would be needed in the knowledge acquisition process, can enhance understanding of a domain. This was also noted in NCC (1987)

> "the building of the system helped the expert to refine and clarify his own knowledge of the domain." (ICI)

> "it has forced him to better organise his existing knowledge of the subject" (British Aerospace)

Furthermore specialists often disagree and using different systems, constructed using the two differing sets of expertise, on a number of test cases and monitoring their performance may help to settle the differences of opinion. It was reported (NCC 1987) that it may highlight the root cause of the discrepancy and so aid further learning

> "the staff were interviewed individually .... brought together as a group to sort out any contradictions" (British Gas)

The use of expert systems can be valuable as an aid to thinking, as has also been reported by NCC (1987)

> "it allows the user to change some of the conditions and values to see what effect it has on the conclusions reached." (ICI)

Turner (1985) reported in a similar fashion about users of Expert Ease, a system which induces rules from examples given by the user. As the knowledge induced from the examples is immediately available in an explicit form, the user can rapidly identify which factors are relevant to solving a problem and the inter-relationships between them. Clement et al (1986) have described how structural relations between objects, but few of the attributes of the objects themselves, are mapped from one domain to another during the learning process. A suitable example is the analogy between the solar system and an atom. In this case only the relations (revolving around, attraction etc.) rather than the attributes (hot, yellow etc.) are transferred to the nucleus and electrons. It is this collaborative use of expert systems that promises to be one of their major future roles, as noted by NCC (1987)

> ".... can be thought of as 'learning with' the expert system" (Universities Superannuation Scheme)

It has been found that if the learner is allowed to browse through the knowledge base in an exploratory fashion then a degree of learning takes place. However if that was all that was needed for full and effective learning to take place, then simply using an expert system in this fashion could become a training mode. The 'learning by browsing' paradigm is understandably more effective for the self motivated learner. From my own experience of browsing through a number of knowledge bases, I have found that I have gained some knowledge of the particular domains, but I would only consider such knowledge to be superficial. This is equivalent to merely browsing through textbooks and it is likely that, in this example, my learning would have been more effective if it had been part of a structured approach. Expert systems have not been widely used in this fashion, but chapter twenty seven contains a discussion of the relevant issues.

Ruthven (1985) pointed out that

> "It must be recognised that learning has important affective and social dimensions which need to be incorporated in any comprehensive model .... it would seem that powerful and detailed computational models of human learning are still some way away."

Hence, not only are there a number of technical and technological issues still to resolve, there are also considerations which need to be given to the environment where the system is to be applied. The quality of learning is determined as much by the learning environment as the ability to learn and issues of motivation, enthusiasm and support should not be dismissed lightly, as any classroom teacher will testify.

MACHINE LEARNING

In the same way that human learning is the key to human intelligence, machine learning is the key to machine intelligence. As noted by Forsyth and Rada (1986), there certainly can be no machine intelligence without machine learning. AI scientists have attempted to build intelligent systems for the last thirty years and Forsyth and Rada (1986) pay tribute to them for achieving as much as has been achieved in the time.

A human expert learns as a result of his experience and thereby improves his performance. An expert system should be able to do the same, but the present state of the art has not reached this point. Goldstein (1979) recognised that representing only the final stage of 'expertise' provides the system with no means of appreciating how that expertise could develop. Knowledge base construction is a time consuming and expensive business and if machines can be programmed to learn and/or improve their performance as a result of experience, then the development costs for expert systems would fall and the number of applications would increase. It would, if developed, be a very sophisticated system, but this would appear, at the present time, to be an ambitious aim. Partridge (1988) suggested that

> "one of the promises of self-adapting systems would be an escalation of the problems of system maintenance .... the development of robust and reliable sophisticated mechanisms of machine learning could reverse this trend."

Present machine learning systems are crude when compared to human learners. Indeed their poor performance merely underlines the sophisticated complexities of the process.

Michie (1986) noted that

> "machine intelligence is not an exercise in philosophy but an engineering project."

McCarthy (1958) suggested that

> "In order for a program to be capable of learning something it must first be capable of being told it."

Machine learning systems offer a way through Feigenbaum's knowledge bottleneck (1979) and a way of synthesising new knowledge, although the knowledge may not be 'new', rather a different perspective on the same knowledge. McLaren (1984) describes Expert Ease, a commercially available system, that uses techniques of induction, a topic discussed in chapter ten.

Can a machine be intelligent ?

Attitudes towards intelligent behaviour by computers have been shaped by a lack of knowledge and understanding of the work that has taken place. A common argument against machine intelligence is that the brain is a living organ and the machine is not. How this compares to concluding that research into artificial hearts is wrong because the artificial one is not living is a question I would put to any artificial heart recipient or their relatives. However, this analogy cannot be carried too far, as I do not see an end product of AI research being the production of a transplantable artificial brain.

Whether a machine can be intelligent is a question that has been argued over for a long time. However it presupposes that we have a clear definition of intelligence and that the only consideration is whether a machine could have it. A machine's intelligence is purely automatic and it can only be the result of what it has been programmed to do. However, is not the same true of humans in that our 'programs' have been developed over years of evolution, rather than developed in computer science laboratories during the past couple of decades.

To concede that machines can exhibit intelligence is to admit that there is a rival in an area previously held to be the sole province of Man. I feel that the basic issue lies in the use of the word 'rival', with machines being characterised as a threat. Being an optimist, I hope that they will be seen as allies, as did Evans (1979)

> "Man has made measurable intellectual progress on his own and it is unthinkable that increased progress will not be made

once Man enlists the help of computers."

The problem of making a machine appear intelligent is a different problem from that of enabling the machine to appear to learn. Furthermore, how to enable a system to learn is inseparable from that of how to represent the knowledge concerned. Sometimes what is learned is stored explicitly as data or facts which may be examined in a variety of ways, or the knowledge may be stored implicitly. In symbolic computation, the manner in which the data is accessed and manipulated is unlikely to be known beforehand and the structure of the program itself may be in a state of flux during its operation. For example, a program in FORTRAN might solve polynomials by testing various values, a LISP program might solve a chess problem by alternately constructing and destroying lists of possible solutions.

EURISKO (Lenat 1981) is an experimental program exploring how it can discover new heuristics. The program is given access to as much information as possible including its own code. It was left running overnight and it was found in the morning to have discovered how to 'cheat'. This might be a warning that such programs although they do what the programmer has instructed, may in addition do things that the programmer didn't intend.

Forsyth and Rada (1986) suggest that there are two basic strategies for machine learning, a 'bottom-up' approach, starting with almost no information and testing what can be discovered. Alternatively a 'top-down' approach, taking an almost perfectly working system, removing a small part of it and investigating ways of automatically replacing the lost information.

Langley (1982) describes SAGE, a system based on a strategy of experimentation. The system learns gradually and in this respect, mimics the incremental nature of much human learning. Langley proposes that the behaviour of the system provides evidence that the following general learning principles play a central role in strategy improvement. A learning system must be able to;

a) generate alternatives (learning by making mistakes)

b) determine when performance has improved or degraded

c) correctly assign good or bad performance to specific components of the performance system

d) modify its behaviour as a result of (b) and (c)

Self (1985) provides one discussion of the issue of student modelling which entails representing, within the computer, a model of the behaviour of the student. Klahr (1976) takes this issue a stage further by attempting the design of a learner. Arguing that learning as problem solving is a process of the amendment of knowledge to reach the desired state. Learning from failure whilst problem solving may make it possible to solve previously intractable problems, although learning from success creates more powerful control knowledge.

Michie (1982a), Michalski et al (1982) and Cohen and Feigenbaum (1983) provide surveys of machine learning programs. Mitchell, Carbonell and Michalski (1986) provide a thorough survey of current research developments in the area of machine learning. Becker (1987) provides an overview of types of learning which can be compared with the summary of human learning situations provided in the previous chapter.

Anderson (1983) and Lenat (1983c), among others, demonstrated that rote learning is not a practical method for most applications because it is based on the premise of learning about each individual experience as it is encountered. This approach means that as there is little likelihood of situations being identical, vast quantities of experiences will need to be stored and subsequently searched to find an appropriate course of action. However, Raphael (1976) argued that rote learning is the most basic activity of every computer system. For example in a database program, the computer 'learns' about each new entry by placing it in a table and memorising that entry and that position. This 'look-up' system is not a particularly sophisticated level of learning. The game of chess has provided many workers with situations for research and experimentation. Even using the most powerful computers available, it is not possible to represent a sufficiently large section of the game to enable the computer to choose the best move, at any one time, simply by looking-up the specific position in memory and selecting the appropriate move. More specialised and generalised knowledge of the particular properties of game situations needs to be applied.

In some game-playing programs, the computer 'learns' a strategy from its opponent and then uses this strategy, when it encounters a similar situation, to make a move. This is learning in an artificial sense and is not necessarily utilising the same processes as human learning. The acid test for AI programs is whether they improve their performance over time.

Using the random rule induction technique, the system generates a rule at random and tests its performance. Successful rules will be retained and combined with other successful rules in the hope of finding new rules. This is an extremely inefficient method. Using an algorithmic rule induction approach, the system constructs a decision tree based upon examples of conditions and outcomes. Such systems need a good training set, covering all possible eventualities and to be able to discriminate between relevant and irrelevant attributes. This method approaches learning from a subtractive, 'top down', paradigm in that irrelevant examples are removed from the decision tree. A major problem is to decide what constitutes 'relevant'. Some means of generalising rules and learning as a synthetic activity was sought by Lenat (1977).

The methods of inductive learning or learning by analogy involve looking for similarities and differences between more than one example of the concept. A different approach, termed Explanation Based Learning (EBL), is deductive rather than inductive and makes greater use of the domain knowledge (Mitchell 1982, deJong 1983). Eary (1987) describes a project that employs this technique of learning.

The aim of EBL techniques is to explain about the example by an analysis of the particular domain and how the relationships link together. As this technique is heavily knowledge-based, it has only been applied to domains where there is a good deal of knowledge about the descriptors of the domain. However, as was described in chapter two, explanation is a very complex business: it is often far easier to do something than explain it. This may relate to the commercial applications of expert systems, as described in Part Two, in that it may be more convenient to use the system to 'do the job' rather than developing a system aimed at 'teaching how to do the job'.

Worden (1988) suggested that the EBL mechanism provides an example of meta-level reasoning

> "On seeing the example, the learner first produces an explanation of it which is a piece of inference using domain knowledge. In order to generalise the explanation, the learner has to reason about how that explanation may be changed and still be valid. This inference about an explanation is a piece of meta-level reasoning, so one could argue that meta-knowledge is a pre-condition for EBL."

The notion of a 'learning apprentice' system (Mitchell et al 1985) has developed from EBL research. It accumulates knowledge through interaction with a 'teacher'. Further developments may result from combining similarity-based learning with EBL and developing, particularly for 'learning apprentice' systems, the ability to learn from mistakes which would be beneficial. Tecuci (1988) describes DISCIPLE, an interactive system which combines learning by analogy, explanation-based learning and empirical learning. Research in this area is at an early stage (Hall 1986, Hammond 1987). Significant progress in explanation theory is likely to have a major impact on the uses of expert systems in training.

Lawler (1985) sees the method of learning by example as being the most effective learning strategy. Teaching by example has similar advantages. The art of choosing good examples is an important problem solving skill. To learn from example it is necessary to know which features of the example are important. The clarification of concepts will be aided by examples which show the similarities and differences between one idea and related ideas. A sequence of examples should start with simple examples and build up to more complex examples which cover exceptional cases. In a pattern recognition system, it is the pattern recognition that provides the complexity. For example in a diagnostic medical system, once the pattern has been recognised and the diagnosis achieved, the choice of treatment will be limited and may only involve the use of a 'look-up table'.

The work of Rosenblatt (1958) and Minsky and Papert (1969) established the method of parameter adjustment, a 'bottom up' approach which involves the system 'homing in' on the best possible answer, by being programmed to adjust internal parameters automatically whenever the computation produces an incorrect solution. This method requires a vast training set to enable the system to cope with a variety of problem situations.

For further discussions of various approaches, see Mostow (1983b), Carbonell (1983b) and Carbonell (1983c) on learning by analogy, Mostow (1983a) learning by taking advice, and Lenat (1977b), Lenat (1983a) and Michalski and Stepp (1983) provide observations on learning by observation, discovery and experimentation.

Walker (1987) discusses issues in automated discovery and identified several promising directions for future work. O'Shea (1987) identified that machine learning systems are limited by their initial heuristics and language, by pointing out that AM (Lenat 1977) had not learned any new concepts since its original publication. There remain major problems still to be faced by workers in this area and O'Shea (1987) suggests that there is little immediate prospect of systems which theorise in a scientific manner.

Current machine learning approaches are inadequate to deal with issues such as adaptation and self-modification which are central to the concept of discovery learning systems (Self 1985). The production of machine learning systems will only arise as a result of long term research projects such as that established by Siemens Research Laboratories and M.I.T. (Computing Techniques 1987).

Scientific discovery usually results from the examination of existing data. The tireless and exhaustive search through this data, by a computer, would seem to offer a reasonable prospect for automated discovery. META-DENDRAL (Buchanan and Feigenbaum 1978; and Lindsay et al 1980) was one of the earliest automated scientific discovery programs. Working within the domains of physics and chemistry, BACON (Langley et al 1983) uses a set of data from previous experiments and employing a technique of heuristic search, seeks constancies and common patterns between mathematical relations. The efficiency of such systems depends upon the lack of 'noise' and the clarity of the data set.

Norman (1985) summarised the differences between the nature of human cognitive systems and those of machines. Klahr (1976) noted that most of the time, human learning does not occur, whereas most AI programs are single-minded in their attempt to learn at all times. Organisms do not have the luxury of this single-minded approach, they must, as identified by Norman (1985), be multiple-minded, data-driven by environmental events. However, humans cannot be completely in this state because there is too much information to process at any one time. Humans select, consciously or sub-consciously, information which at that moment in time seems interesting or important. This conscious or sub-conscious information processing, which, although not needed by all cognitive processes, is a major difference between animate and inanimate cognition.

There would appear to be two areas where developments in machine learning may bear fruit;

a) The short-term practical issue - If machines can be programmed to learn as a result of experience, then the problems associated with knowledge acquisition, discussed in chapter ten, would decrease and there would be a probable increase in the scope of commercial applications.

b) The long-term theoretical issue - The knowledge gained in (a) may provide valuable information to help our understanding of human learning. However this is not guaranteed and, as noted in chapter four, it may be that there are no common basic principles between different intelligences.

As has been discussed in this chapter, present machine learning systems are crude when compared to human learners and many problems remain for research workers in this area.

THE MAN-MACHINE INTERFACE

The man-machine interface is the system developed to enable human-computer interaction. An expert system which is capable of making 'expert' decisions will not necessarily be successfully implemented unless attention has been paid to the user interface. Faced with two software packages which perform similarly, the purchaser will often choose the one which 'feels right'. More often than not, this is a function of the user interface.

An 'intelligent' interface between systems and end-users is a reasonable aim, but there is not universal agreement as described by McCracken and Akscyn (1984). Innocent (1982), Bundy (1984) and Rissland (1984) variously describe some of the problems and issues in this area. The main reason for the current lack of attention to user interfaces is the fact that they occupy a large proportion of the code. Goodall (1985) cited a study that suggested that in a typical system, 8% of the code is inference engine, 22% knowledge base and 44% user input/output. Berry and Broadbent (1986) noted that much of the emphasis of expert system research has been aimed at developing working systems, the user interface being seen as

> "something to be tagged on the end."

The desirable properties of such interfaces are currently unknown, although Berry and Broadbent (1986) provided a list of user interface considerations;

a) excellent decision making by itself is not enough, systems must be good consultants as well as problem solvers. The user may wish to question the system's decision.

b) most, but not necessarily all, applications warrant sophisticated flexible dialogue characteristics

c) good explanation features are vital. (the style of the explanation dialogue is related to the type of end-user)

d) few systems incorporate dynamic user models

e) the inclusion of a natural language interface may be a long term aim, but this area is largely at the research phase and it is not feasible to add-on a natural language interface to an existing expert system

f) the end user must be considered at the beginning of the project

g) discipline is needed in the use of expert systems if novice users are not to be overloaded (i.e. assume the user knows nothing)

h) designers will need to prevent users overestimating the abilities of the system

i) user modelling, so that the system can cope with a wide variety of users

j) the handling of probability which humans tend to process in a qualitative rather than numeric fashion

Adaptation is a key word, an adaptive interface will change the behaviour of the system in response to the user. A tutoring program will attempt to adapt the behaviour of the user. Regardless of whether these changes would be successfully achieved, it is certain that the provision of a quality man-machine interface is of prime importance in any system.

In the 'real' world, experts use knowledge about their clients to decide what advice to give and how to present that advice. Kidd (1985) also identified several other important types of knowledge;

a) knowledge about the underlying causal mechanisms in the domain

b) knowledge about the decision making methods used by the system

c) knowledge about how good explanations are constructed (this is of particular importance in educational systems)

Kidd also noted that

> "if a system is to be responsible
> for complex decision making and

> giving advice, then it is vital
> that there is compatibility between
> the user's model of the problem and
> the system's."

The system may 'fail' if the knowledge base is inadequate or if the system and the user misunderstand each other. Diaper (1986) pointed out that there is considerable potential for such misunderstanding.

Little is known about the cognitive aspects of system users and Norman (1984) has suggested that in addition to communicating between user and machine, another role for the interface could be to establish the intentions of the user. Diaper (1986) suggested that such interfaces may assist in preventing any cognitive misunderstanding between people and machine intelligences. In addition, misunderstandings may also arise as a result of the fact that natural language includes vague concepts, such as body language, which a machine would be unable to interpret.

Verbal natural language is the usual form of human communication, but it must be questioned if it is also the most appropriate form for man-machine communication. The term 'natural language' presents problems in that some workers including Sparck Jones (1984), Murray and Bevan (1984) and Richards and Underwood (1984) perceive the dialogue being comparable to how people talk to each other over the telephone. This supposes that expert systems have a similar status to human users. At this point it is interesting to note that although reactions vary, a feeling of unease is often reported by users of telephone answering machines. Bachtin (1984) and Newell (1984) provide a contrasting viewpoint as they perceive computers not as equals, but as tools. I am very much in sympathy with the latter viewpoint.

Very few systems have been implemented with any natural language capability and few organisations have invested in the necessary research. This is not surprising: natural language research is a very complex area. For an appreciation of the issues involved, see, for example, Boden (1977), Smith and Green (1980), Sparck Jones (1984), Wallace (1984) and Johnson (1986). A full natural language dialogue is unnecessary for communication with a computer, the level of sophistication will depend upon the application and the intended end user. For example, MYCIN (Shortliffe 1976) works in a narrow domain and the number of responses possible at each stage of the consultation is limited. Hence there is a dialogue, which the authors of the system termed 'doctorese'. It may be possible to invent or adapt similar sub-sets of language (eg as are used in computer adventure games) for use in other suitably restricted domains.

It was noted by Kemp et al (1988) that

> "The majority of expert systems work as though the user is talking to the expert on the telephone and therefore all the exchanges are

verbal or textual. This approach fails to utilise the important human sense of sight. Visual information speeds up the process of recognition and at the same time helps to keep the user interested and alert."

Gill (1986) identified that there is a danger that the development of machine-centred systems, which are only capable of depositing limited quantities of knowledge onto the learner, will restrict the expansion of human skills and experiences. This developing trend of transfer of intelligent activity from human to machine will have consequences which must be considered. Smith and Green (1980) suggest that in the development of expert systems, there is a philosophy of shared responsibility, noting that automation doesn't remove the human from the system or turn them into automatons. As Forsyth (1984a) observed, the automating of human knowledge can only be achieved by stripping it of such human traits as creativity, fuzziness and ingenuity.

> "does anyone these days admire someone who can dig a hole or paint a car quickly ? We shall soon feel the same dullness about brainwork."

Might this be a sign of what is to come ? Remember Marvin the robot (Adams 1978) who had a brain as big as a planet and yet was chronically depressed.

Much of the wider discussion about expert systems and their applications concerns technical matters. However, when considering educational applications, the social context of the learning environment must be taken into consideration. Consider the difference between using a piece of CAL, as a training aid, in a very restricted technical domain and the conventional school-based learning situation. The learning, or arguably the mere transfer of knowledge, in the former case can almost take place in vacuo, whereas the latter is a complex social situation and there are no AI techniques which currently offer the capability of dealing with these issues.

Jacob et al (1986), provides a broad comparison between those components which humans use to make decisions and those which are commonly found in expert systems. The comparison illustrates why it is difficult to model human decision making with a computer, noting that human factors, such as intuition, creativity and motivation, are absent from machine systems. Consistency is a factor which is often used as part of the argument for using machines, although it can be argued that humans usually make good decisions because of the diversity of factors which contribute to the process. Although the machine would make a consistent decision, given identical problem parameters, it would not create a new solution, as may be achieved by human decision makers. Under these circumstances, the benefit of using a man or a machine would depend upon the specific application. In a simple application such as room heating, a thermostat could be used on the basis that

IF room temperature = hot THEN switch off heating
IF room temperature = cold THEN switch on heating

Note that Searle (1984) argued that on this basis, the thermostat had beliefs. I refute Searle's suggestion, preferring the counter argument that a human could follow the above rules, switching off the heating when the room 'felt' hot, whereas the thermostat would need to work to a precise definition of what constitutes 'hot'. However, as the application increases in complexity, so does the problems of deciding whether to use a man or a machine.

SOCIAL IMPLICATIONS

Little is new, or so it would seem from this quote, reported by Howe (1987), of an observation made about Babbage's Analytical Engine, by Lady Lovelace over a century ago

> "In considering any new subject, there is frequently a tendency first to overrate what we find to be already interesting or remarkable: and secondly, by a sort of natural reaction to undervalue the true state of the case when we do discover that our notions have surpassed those that were really tenable."

It is possible to discuss the applications of advanced technology within education but it would be remiss to neglect to discuss the many social implications. The impact of AI is real and is growing, and any evaluation of the potential and limitations of expert systems must take full account of the human dimension, which cannot be disassociated from the technical considerations. The danger of the technology was summarised by Kowalski (1987) who reminded us that it is easy for us to let experts (eg doctors, accountants) take over our decision making and intimidate us with their knowledge.

> "If humans can intimidate humans, then computers will be able to intimidate humans too; and they will do so, if we allow the enthusiastic technologist to have his way. The technologist will happily design computers to do more and more of our thinking for us."

The arguments concerning the social implications of AI seem to fall into two camps. There is a 'utopian' view as expressed by Evans (1979), McCorduck (1979) and Boden (1984). On the other hand there is the 'no good will come of it' view as expressed by Weizenbaum (1976), who questions the morality of it, and Dreyfuss (1979) who suggests that it is mistaken in principle. While noting the arguments of both Dreyfuss and Weizenbaum, I am a subscriber to the 'utopian' philosophy, provided that there is an awareness of the technical limitations and that the social concerns, for example those expressed by Gill (1986), are included in any programme of development.

The limitations of systems will exist regardless of computer processing power or memory capacity. Simons (1983a) suggested that

> "the limitations of expert systems would be particularly prevalent in micro-based systems and that the end users of such systems, possibly the most inclined to trust the deliverances of clever low-end systems, would be most at risk."

Technology is providing information that is more easily accessible. However, the law of economic diminishing margins may apply to this increase in available information. If apples cost 100p per Kg, then people will buy them, if the price falls to 50p per Kg then they may buy 2Kg, but they will not buy 3Kg even if the price falls to 25p per Kg. It may be that the technology is bringing a glut of information. When we have a glut of apples a proportion of them are dumped, will the same happen to a glut of information ? Technology may again provide the answer by assisting in the increasing amount of information processing that will take place.

One consequence of the introduction of expert systems may be that professionals will experience the feelings of employment insecurity that manual factory workers have felt as plant automation has increased during the last decade. This will depend upon the extent to which such systems are used as intelligent assistants or as replacements for human operators. However, Michie (1986) suggested that

"the indications are that as soon

as brain workers learn to use the new facilities, their work will be enlarged and enriched by the new possibilities which become available to them."

If through the use of expert systems, the power of an expert is available to the learners, we must be aware, as was indicated by Evans (1979), of the possibility of passing to the learners, via the inherent (implicit or explicit) content and structure of the knowledge, particular values, prejudices or viewpoints of the author of the system.

The impact of AI technology on employment is difficult to quantify with any degree of accuracy. Stonier (1983) has argued that we are in the middle of an information revolution which is similar to the earlier agricultural and industrial revolutions. IT-caused unemployment is structural and is largely irreversible until the structure is changed, as was the case with the earlier revolutions. The technology itself is neutral and, as noted by Dunn and Morgan (1987), although the development may have caused a problem for society, IT doesn't make decisions about how society is organised. Whittet (1987) argues that IT is not guilty of causing unemployment, although it may have a small effect on the levels of employment in offices. Commercial innovation occurs because it is economically feasible to use it. The impact of the technology will not be simply redundancy, but a restructuring of demand for some types of labour. Here there are considerable implications not only for education, but for society as a whole.

A study of the rules contained within MYCIN (Shortliffe 1976) shows that they are shallow rules which are associated with cause and effect, covering most of the simple cases. Under similar circumstances to these, the use of the expert system would be justified, perhaps, on the grounds of saving the consultant's time. However, the expert human consultant would be required if the situation was different from those covered by the rules.

There are a number of ways in which an expert system can assist an expert and his colleagues to improve their performance. However the question arises as to whether in the long term such systems pose a serious threat to the livelihood of those professionals and experts whose knowledge they contain. Clearly if replacement rather than assistance were perceived to be the most likely development, then those threatened (who constitute a considerably more powerful group in society than those made redundant through the automation of manual jobs) would inevitably react in a negative fashion. The reality is likely to mirror development in all earlier phases of mechanisation and automation in that there will be an initial period after the technology becomes truly useful during which there will be replacement of some human activity by machine. However, Michie and Johnson (1985) doubt the efficacy of current systems as replacements for human experts. Additionally, Beynon (1985) noted that CAL programs should be developed to exploit their adjunctive role rather than as a replacement of teacher functions. After this time the additional productivity thereby gained will increase economic activity and this growth will create demand for human resources at a higher skills level. Hence it is envisaged that the replacement will be of 'lower level' activities and thereby freeing more human time for more interesting, challenging tasks. This will have an impact upon national educational and training needs.

Pearson (1984) provides a positive vision of what the future holds, identifying the fact that developments offer the opportunity to change many of the 'professional' boundaries and create a new professional and managerial revolution, which will obviously have profound social implications. Additionally, there are associated educational implications for training about, and with, expert systems and the extension of self education.

One of the characteristics of expert systems, as discussed in chapter two, is the explanation facility. Ennals (1986a) discusses how this will affect our relationships with 'experts' and the whole social and economic infrastructure. The political consequences of expert systems merit further study because, as noted by Stevens (1984)

> "when knowledge is available to
> both sides of a social conflict
> then the power relations between
> the two sides must also change."

Interaction with computer systems does not involve the usual social consequences and may facilitate the exploring of ideas which may be inhibited by a human presence. Pateman (1981) identified that

> "I wonder what it <the computer>
> will do if I say this ?"

is significantly less threatening than

"I wonder what he <the teacher>
will do if I say this ?"
(my brackets)

Jones (1980) noted that

"the greatest barrier to the
socially responsible application of
microelectronics is ignorance and
fear engendered by that ignorance"

and Michie (1982b) suggested that

"the greatest social urgency
attaches not to extending automatic
processes but to humanising them."

There may be wide divergence between what is technically possible, now or in the future, and what society would consider desirable. For example, Ennals (1986) argued strongly that socially useful research should be undertaken rather than spending billions of dollars on military applications. The emphasis should be placed on investigating how to use the information that IT can provide.

SUMMARY AND CONCLUSIONS

The 'Fifth Generation' computer revolution remains some distance away and it could well be the end of this century before general purpose intelligent systems have been developed. Indeed, AI research activities are not searching for the 'Holy Grail', as Davis (1984) suggested

> "There seems to be no philosopher's stone, no single clever trick that will solve problems for us across all problem domains."

The present technology is largely based on research done in the early seventies. Anderson (1985) recommends that

> "the researchers stop pretending that research bears on current practice"

In the meantime progress will be made on some of the fundamental issues (eg machine learning, knowledge representation, language understanding). However, some advances have already been made and expert system technology represents one of these, but they are, as was discussed earlier, specific, rather than general, applications.

The recent development of the commercial applications of expert systems has made advances in technological efficiency, but not necessarily to scientific understanding. I feel that a likely future scenario of commercial development is for expert system technology to be embedded in other software, so that expert functions will develop in larger software projects.

There is a view that computer technology can be used as an educational aid that is slightly more sophisticated than a video-recorder. Any subscribers to this view are likely to underestimate the radical effect that advanced technology will have on education. The vision of powerful computer systems linked over wide networks to interactive videodiscs and laser printers may be termed futuristic, but the technology is already here and it will continue to develop. It may be that in the final analysis, the issue is less a technological one, but rather a social one. As noted by Heaford (1983)

> "the computer has power to change
> the learning process beyond
> anything we have approached so far,
> but the environment must change to
> accommodate it."

AI will have a considerable effect upon CAL and there will be a widening of the scope and range of topics where it is used. AI-based CAL is going to be capable of providing a far deeper representation of the knowledge than that used so far in CAL. It is anticipated that successful attempts will be made to improve the competence and capabilities of traditional courseware by embedding expert systems in them or by the incorporation of sophisticated knowledge representation techniques. Another improvement which AI research offers is the production of more 'intelligent' adaptive user models. However, there remains the danger of CAL containing an implicit view of education as a product, rather than as a process. Sleeman and Ward (1988) suggested that many computer-based courseware systems have not been a lasting success because they were both technologically and pedagogically premature. During this decade technological advances have been made and although many points of detail remain unresolved, we also now have a much better appreciation of the general pedagogical issues that are involved.

The curriculum issues were identified by Fletcher (1983) who also noted that

> "the greatest change of all is in the response of the pupils. Pupils are displaying this new-found enthusiasm
> .... this enthusiasm should be turned to good effect without delay."

However, Campbell and Millar (1984) clearly established that the enthusiasm alone was not enough and there might be an element of 'line drawing'

> "the question of the extent to which we are justified in going along with the pupils' interests, and where we might need to draw some sort of line, and what the criteria for such line drawing might be. This question is not an easy one and individual responses may well differ significantly."

Experts use judgement as well as logic in arriving at their conclusions. It may be possible to capture expertise as a set of logical rules, but programming judgement is not so simple. Searle (1984) argued that systems based on logical arguments may be restricted to 'low-level' applications. As a result of publicity and indeed the name itself, user expectations of expert systems are high, these systems are unlikely to be 'intelligent', they may simply make a better job of logical deduction than 'experts'.

Bundy (1987) noted that

> "most of the interest in expert systems is not because of their proven capability, but because of their potential
> .... all new technological advances have the potential for both good and bad applications
> .... we need to be aware of both the potential benefits and dangers of expert systems"

As an example of one such danger, Evans (1979) warned that such systems could contain and express the prejudices of their authors.

Michie (1982b) suggested that a trainee chemist using a 'chemist's assistant' could become as valuable to a chemical company as a senior chemist. Similarly a junior houseman using a 'doctor's assistant' could become as valuable to the hospital as a senior consultant. Many of the present systems are expert systems for experts, more attention needs to be paid to the naive or novice end user.

Stonier (1984) cautioned about the dangers of compounding complexities by applying to already complex situations which are poorly understood, new levels of complexities which may also be poorly understood. The possible consequence of this may be the production of new levels of uncertainties associated with the delusion of higher levels of accuracy. This may further lead to increased unquestioned acceptance of system decisions which would be a retrograde step.

The application of expert systems within education should be viewed as part of Man's efforts to use machines to augment mind as well as muscle. The earlier industrial revolution brought undoubted benefits and associated disadvantages and a wide range of social implications. The 'expert system revolution' is likely to have equally wide ranging implications. Nevertheless I remain optimistic for the future.

What can we learn from commercial applications ?

The introduction of expert system technology into the commercial arena has been surrounded by hype and razzmatazz to the extent that many products have appeared with the 'expert' or 'intelligent' label which on closer examination have proved to be no more intelligent than any other software package. As an aside, there is the school of thought that this, depending upon the parameters of your definition of 'expert system', as discussed in chapter two, is a truism anyway. People's expectations have been raised and then shattered when the software failed to match expectation. In such cases they are not keen to be deluded again. Failed attempts to develop expert system applications are not often publicised, but mistakes can be productive. The secrecy factor, identified by d'Agapeyeff (1984a), as it relates to commercial organisations, should not apply to the world of education.

It is inappropriate to analyse the potential educational applications of expert systems in the same way that commercial applications were analysed in Part Two. The reasons for commercial developments, as detailed in chapter fifteen, are not the same reasons for interest in their educational applications. Nevertheless commercial profit may indeed be a prime motive for the increased use of expert systems in a training capacity, particularly as companies come to realise that the skills and knowledge of the workforce is a company asset, albeit one that is difficult to show on a balance sheet. Although chapter twenty five proposed some curriculum analogues with commercial applications, the 'islands' analysis, reported in chapter fourteen, doesn't help here as the systems listed in chapter twenty seven come from a wide variety of domains. The exciting potential of expert system technology is that it could be employed as a tool across curriculum areas.

It is difficult to supply empirical evidence on the efficiency of learning with expert systems, which has not yet been fully evaluated, and as noted by Briggs (1987)

> "using expert systems in education could not yet claim that their use produces better results or brighter students."

However, Self (1987a) argued that

> "IKBS is important to education because of its focus on the development of individualised micro-theories of learning and teaching and on the role of psychologically-based representations of subject

> knowledge. While these are considerable contributions, they do not and arguably should not be thought of as 'theories of education'."

Education will continue to be the poor relation of military and commercial interests and Brough (1987) noted that

> "little of current IKBS research is targetted directly at applying KBS to education. The current emphasis appears to be collaborative projects aiming to develop products for industry. Fortunately many of the research themes turn out to be relevant to the development of support tools for the education environment
> .... its application to schools can support less tangible, but nevertheless vital, improvements in terms of quality of education"

Although Dixon (1988) commented that there may be theoretical limitations (especially regarding the level of knowledge held by the system) to the effectiveness of using expert systems in training, they are already being used for this purpose. The explanation for this apparent paradox is that training does not require instruction per se.

There may be, in the fullness of time, a radical reappraisal of education as we think of it today. I believe that this could be felt most in areas of Further and Higher Education as commercial companies may boost their own in-house training, coming to rely less and less on the Colleges. A further factor may be associated with the development of an increasing range of 'open' and 'distance' learning opportunities. Flexibility will be the keyword of the use of advanced computer technology in education, resulting in systems which are more responsive to the needs of the individual.

The 'bottom line' and the future ?

Making predictions about the development of new technology is an uncertain science. In the late 1940s it was believed by IBM that the world demand for computers would be a handful of large scale systems. If predicting technological change was difficult, attempting to predict the changes to education over the next couple of decades is even more tricky as education is moulded or buffeted by political and social pressures.

a) Expert systems will become increasingly important in the future as they become both easier to use and more readily available.

b) The technology is already here, expert systems are being used in increasing numbers to perform different tasks in a variety of commercial domains.

c) Industry and commerce are interested in completed working systems, whereas education is more concerned with the process of developing systems.

d) Already expert systems are being used as training aids; training about expert systems, with expert systems and through the use of expert systems.

e) The construction of intelligent tutoring systems is a much more complex exercise than developing expert systems.

f) The most exciting area for potential development is in the 'pupils as knowledge engineers' learning through building systems paradigm.

g) Learning through the use of expert systems cannot be considered in a vacuum isolated from the various social conditions of the learning environment.

h) If it can be shown that expert systems can provide efficient and cost effective training at the place of work, then education may be 'released' from vocational considerations.

i) As access to information increases in importance, the application of expert systems as intelligent front ends to databases or as part of intelligent retrieval systems will provide a vital developing role.

j) The potential that expert systems could make everyone his own expert has considerable implications for education.

It will take time and it will be expensive, but the challenge is there for education to harness the power. As the Chinese proverb puts it, the longest journey begins with a single step. Several steps have already been taken.

APPENDIX 1

Forward and backward chaining

To find a solution to a problem, an expert system uses one of two general strategies - forward or backward chaining. Forward chaining is a bottom-up data-driven strategy that requires the user to volunteer facts. A forward chaining system starts with a set of facts which describe the characteristics of the problem and it then applies productions or logical inferences whenever their conditional parts are satisified by the current facts. This approach forces the system to search forward from the original contents of its knowledge base, making inferences as it goes, in a gradual synthesis of a solution. This method works best for problems with many solutions or where a goal must be built.

In contrast a backward chaining system works top-down from the goal through various sub-goals. This goal-driven approach works best when the goal is known and the number of outcomes is small. A backward chaining system starts with a specified goal and works backwards from that goal trying to find a sequence of rules that it can apply to infer the validity of the goal from the initial facts in memory. Hence its approach is one of a gradual analysis of a solution by the analysis of parts of the problem as they arise and because of these characteristics, it is commonly used for diagnostic applications.

To show how the two systems would work, the following examples are given. The two approaches are being applied to finding out whether X is true given the following rules;

Rule 0 : system has done its job if we prove X

Rule 1 : X is true if [A and B]

Rule 2 : X is true if [A and C]

Rule 3 : C is true if [E and D]

We start with the knowledge that A, D and E are true.

Backward chaining

(Goal) : Need to prove X

(Rule 1) X if A and B

    (Goal) : need to prove A

    A is true

    (Goal) : need to prove B

    (Rule 0) : no use

    (Rule 1) : no use

    (Rule 2) : no use

    (Rule 3) : no use

    no rule proves B true

    B is false

(Rule 1 fails)

(Rule 2) : X if A and C

    (Goal) : need to prove A

    A is true

    (Goal) : need to prove C

    (Rule 0) : no use

    (Rule 1) : no use

    (Rule 2) : no use

    (Rule 3) : C if E and D

(Goal) : need to prove E

E is true

(Goal) : need to prove D

D is true

(Rule 3 succeeds) : C is true

(Rule 2 succeeds) : X is true

(Goal proved , X is true) : stop.

<u>Forward chaining</u>

Rule 0 : X not proved (fails)

Rule 1 : A true, B not true (fails)

Rule 2 : A true, C not true (fails)

Rule 3 : E true, D true (succeeds)

-> sets C true

Rule 0 : X not proved (fails)

Rule 1 : A true, B not true (fails)

Rule 2 : A true, C true (succeeds)

-> sets X true

Rule 3 : E true, D true (succeeds)

-> sets C true

Rule 0 : X proved (succeeds) -> stop

Certain assumptions have been made about the way in which the rules are called and applied (that they are tried in cyclic fashion and in the order in which they are written). These are not universally used, but they give enough of an example to contrast the two methods. There is a parallel with the 'top-down' and 'bottom-up' approaches of program design. It should also be clear that if given the proper set of rules, forward chaining systems may be effectively forced to backchain and vice versa. However the distinction between the two approaches is important in that a backward chaining system allows AND parallelism more easily. The latter being a form of parallel processing where an interpreter takes advantage of the existence of multiple terms in a clause, or in a rule's test conditions, to establish each term in parallel.

For example, a medical diagnosis system will reason backwards in order to confirm or deny a given diagnosis. An advisor system would need to reason forward from the data to a conclusion tailored to a specific patient.

APPENDIX 2

The Turing Test

In 1950 Alan Turing wrote an article entitled 'Computing Machinery and Intelligence'. In it he considered the question of whether machines can think, by replacing the question with another. I have quoted, below, from his paper, as published in Feigenbaum and Feldman (1963).

"The new form of the problem can be described in terms of a game which we call the 'imitation game'. It is played with three people, a man (A), a woman (B), and an interrogator (C) who may be of either sex. The interrogator stays in a room apart from the other two. The object of the game for the interrogator is to determine which of the other two is the man and which is the woman. He knows them by labels X and Y, and at the end of the game he says either 'X is A and Y is B' or 'X is B and Y is A'. The interrogator is allowed to put questions to A and B thus:

C: Will X please tell me the length of his or her hair ?

Now suppose X is actually A, then A must answer. It is A's object in the game to try and cause C to make the wrong identification. His answer might therefore be:

'My hair is shingled, and the longest strands are about nine inches long.'

In order that tones of voice may not help the interrogator the answers should be written, or better still, typewritten. The ideal arrangement is to have a teleprinter communicating between the two rooms. Alternatively the question and answers can be repeated by an intermediary. The object of the game for the third player (B) is to help the interrogator. The best strategy for her is probably to give truthful answers. She can add such things as 'I am the woman, don't listen to him !' to her answers, but it will avail nothing as the man can make similar remarks.

We now ask the question, 'What will happen when a machine takes the part of A in this game ?' Will the interrogator decide wrongly as often when the game is played like this as he does when the game is played between a man and a woman ?

These questions replace our original, 'Can machines think ?'"

APPENDIX 3

The Alvey IKBS Community Clubs

The following table shows details of the 9 Alvey IKBS Community Clubs and their contractors.

Acronym	Members	Industry
ALFEX	23	Finance
ESI and Helix		
ARIES	24	Insurance
Logica		
DAPES	14	Data Processing
Expertech and NCC		
EMEX	14	Economic modelling
ESI and the Henley Centre for Forecasting		
PLANIT	16	Planning
Istel and Systems Designers		
QSES	13	Quantity surveying
University of Salford		
RESCU	25	Real-time
Systems Designers		
TRACE	9	Travel
Wootton Jeffreys, Software Sciences and University of Leeds		
WIESC	14	Water
Software Sciences and University of Surrey		

Both EMEX and RESCU are aiming to produce commercial systems and continue the work initiated by the Clubs.

APPENDIX 4

Expert Systems database

The following list contains details of systems referred to in the text. These details have been extracted from the database which was analysed in Part Two. It was built up over a period of some eighteen months starting in January 1986. The database does not represent, nor was it intended to be, an exclusive catalogue of expert systems, but it does appear to be representative and comprehensive. I did not deliberately go out 'expert system hunting', but added their details as I came across them in the academic or commercial press. The reports of some systems were too vague or sparse to warrant inclusion, but the database grew to over 800 systems and towards the end, I became more selective about what should or should not be included. A bit of selective weeding to remove duplicates reduced the final total in the database to 785 systems.

It was only possible to verify the details of the systems included in the list, by reading about a system from two different sources. I am therefore at the mercy of the various authors when it comes to classification of the systems and any misinterpretations which may have occurred are entirely my responsibility.

The entries for each system are as follows;

A five character acronym, usually invented by the developers or where one did not exist I took the liberty of giving it one.

The domain according to my classification.

Details of the task performed by the system.

A more full name of the system. Some systems are referred to by different names. For example Professor John McDermott refers to his configuring system by its original name of R1, but DEC call it XCON.

A reference to the system. In some cases there were just a few lines about the system, in other cases (eg MYCIN) there have been large numbers of books written about them. My reference, therefore, is a source of information about a system and is not necessarily the definitive paper.

Technical details of the system including the language and machine used and the country of origin.

The task, again according to my classification.

A date giving an indication of when the system was under development. In many cases this spread over several years as the protoype became a working system which was then refined and further improved. For example XCON, arguably one of the most successful commercial applications, has reached the stage where it contains 6000 rules and it is now being rebuilt. It is accepted that the system is never going to have all the knowledge it needs, even though at the moment it is reported to know about 20% of DEC's components (Computing 1987).

In my original database, I included details of the status of the system eg was it operational or a research prototype ? However as the database grew in size, this entry became more and more meaningless, particularly as it became impossible to tell if the author of the report took 'operational' to mean 'the system actually worked' or that the system was in routine commercial use. Therefore I dispensed with this field.

AFD
ENGINEERING
system allows the accumulation of
experience-based knowledge into an expert
system to solve the problem of 'old-timers'
declining in numbers
AUTOMATIC FORGING DESIGN
Smart and Langeland-Knudsen (1986) p6
developed at Battelle Columbus Laboratories
USA
DESIGN
1984

ALFEX
FINANCE
a 'company health assessor' to give advice on the financial well-being of companies
ALVEY FINANCIAL EXPERT SYSTEM CLUB
Hewett, Timms & d'Aumale (1986)
developed by Helix and ESI UK for the Alvey Alfex Club
ADVISOR
1985

AM
MATHEMATICS
system tries to discover new mathematical concepts from old ones
ARTIFICIAL MATHEMATICIAN
Goodall (1985) p64
written by Lenat at Stanford University USA
PREDICTION
1975

APE
EDUCATION
system interactively plans academic courses to meet the student's remaining degree requirements while being sensitive to his strengths and preferences
ACADEMIC PLANNING ENVIRONMENT
IEEE Transactions in Education vol E-29 no2 p120
written by Golumbic et al at Bar Llan University, ISRAEL using Prolog
PLANNING
1986

APRES
COMPUTER-Software
analyses QUDOS crash dumps
ALAN PRIDAY REPLACEMENT EXPERT SYSTEM
Computer Weekly 4/9/86 p34
developed by Ferranti UK
INTERPRETATION
1986

ARIES
BUSINESS
two projects have been developed; fire risk assessment and equity investment advisor
ALVEY RESEARCH INTO INSURANCE
Jones & Davies (1986) p130

developed as part of the Alvey ARIES Club UK
ADVISOR
1986

ATTEN
MEDICINE
system critiques a doctor's plan for
anaesthetic management and as such does not
TELL the doctor what to do, but acts more as
an aide-memoire
ATTENDING
Smart and Langeland-Knudsen (1986) p18
developed by Miller at the University of Yale
and implemented in LISP USA
PLANNING
1982

BACON
PHYSICS
the system induces general physics laws from
empirical data
BACON
Machine Intelligence 10 1982 p269
developed by Langley at Carnegie-Mellon
University USA using OPS2
PREDICTION
1979

BNAES
LAW
an advisor system covering a section of the
British Nationality Act
BRITISH NATIONALITY ACT EXPERT SYSTEM
Simons (1983a) p80
a 150 rule knowledge base developed at
Imperial College UK
ADVISOR
1985

BUGGY
EDUCATION
designed to diagnose 'bugs' in simple
procedural skills
BUGGY
Sleeman and Brown (1982) p157
developed by Burton USA
TUTOR
1982

CANSE
MEDICINE
system searches the MEDLINE database for
cancer therapy literature
CANCER SEARCH
ASLIB Proceedings 36-5, May 1984 p229
written at Huddersfield Polytechnic, UK in
PASCAL and PROLOG on a time-shared Prime 750.
The program is over 3000 lines
PROGRAMMING
1984

CASNE
MEDICINE
performs diagnosis in the domain of glaucoma
CAUSAL ASSOCIATIONAL NETWORK PROGRAM
AI Handbook vol 2 p193
developed by Weiss et al at Rutgers
University USA
DIAGNOSIS
1977

CASTE
ENGINEERING
addresses a particular diagnostic feature by
looking at malfunctions in industrial
sandcasting
CASTER
IEEE Software March 1986 p6
developed by Westinghouse Corp and Stanford
University USA using HERACLES
DIAGNOSIS
1986

CHEES
AGRICULTURE
quality control support system in the
production of cheese
CHEESE
Hewett, Timms & d'Aumale (1986)
developed by CAP Sogeti for Institute
Francois de Gruyere and Ministry of
Agriculture FRANCE
CONTROL
1986

CHEMG
COMPUTER-Software
to aid the use of a computer aided molecular
modelling system called CHEMGRAF

CHEMGUIDE
Expert Systems User Jan 1986 p16
developed by Chemical Design, Oxford UK
PROGRAMMING
1985

CHEST
MEDICINE
diagnoses chest pains for incoming patients at casualty wards
CHEST PAIN DIAGNOSIS
Computer Weekly 21/8/86 p56
developed by Peter Emerson of the Royal College of Surgeons UK
DIAGNOSIS
1986

CLOT
MEDICINE
a blood disorders program which identifies the presence and type of blood clotting disorder
CLOT
Alty and Coombs (1984) p94
developed using EMYCIN as part of the Stanford University USA MYCIN project
DIAGNOSIS
1981

COALS
COMPUTER-Software
a knowledge based interface to bibliographic databases in coal technology
COALSORT
IEEE Expert Spring 1987 p39
developed at Carnegie-Mellon University USA
PROGRAMMING
1987

COMPA
ENGINEERING
aids in the maintenance of a telephone switching system.
CENTRAL OFFICE MAINTENANCE PRINTOUT ANALYSIS AND SUGGESTION SYSTEM
Expert Systems vol 2-3 July 1985 p112
implemented using KEE and Interlisp on a Xerox Lisp machine. USA
MONITORING
1985

CPSFE
COMPUTER-Software
an intelligent front end to a contour plotting software package
CONTOUR PLOTTING FRONT END
Hewett, Timms & d'Aumale (1986)
developed for Statoil NORWAY by Computas Expert Systems using InterLisp-D and Loops
PROGRAMMING
1986

DENDR
CHEMISTRY
system generates the structural isomers of a given chemical composition and eliminates those structures which are chemically impossible
DENDRAL
Machine Intelligence 4 1969 p 209
developed at Stanford University USA
PREDICTION
1967

DEPAM
COMPUTER-Software
DIAGNOSIS using EMPIRICAL, PROPOSITIONAL and ANALOGICAL MODELS
IEEE Software March 1986 p50
project at the State University of New York USA, to develop a framework for a generic diagnostic expert system
PROGRAMMING
1986

DHSSD
LAW
system is planned to browse through DHSS social security regulations, advise on entitlement, policy, help in form completion etc.
DHSS DEMONSTRATOR
Fox (1983) p225
being developed by a consortium as an Alvey Demonstrator project UK. Using APES and PROLOG
ADVISOR
1985

DIAEX
COMPUTER-Software
acts as an advisor in the use of SPIDER, an image processing program
DIA-EXPERT
Expert Systems vol 1-1 1984 p55
developed by Tamura and Sakaue of ETL JAPAN
ADVISOR
1984

DIPAD
ENGINEERING
interprets data from the analyst's dipmeter tool
DIPMETER ADVISOR
Computing (mag) 14/3/85 p15
being field tested by Schlumberger USA
INTERPRETATION
1985

DRILL
GEOLOGY
diagnoses the problems of sticking and dragging that can occur during oil drilling. System also recommends procedures for releasing stuck drill bits.
DRILLING ADVISOR
Computing (mag) 14/3/85 p15
developed by Teknowledge Inc. and Elf Acquitaine using INTERLISP USA
ADVISOR
1983

EDAAS
LAW
helps screen requests for the disclosure of information (specifically data on toxic chemicals) under the USA Freedom of Information Act
EDAAS
Expert Systems vol 2-2 April 1985 p72
developed by the USA Environmental Protection Agency
ADVISOR
1985

ELAS
COMPUTER-Software
advises on the use of INLAN, a complex set of programs for interactive well-log analysis

EXPERT LOG ANALYSIS SYSTEM
Goodall (1985) p60
written at Rutgers University for Amoco USA
PROGRAMMING
1982

EPX
COMPUTER-Software
system searches for documents in the Chemical Abstracts database serving as an intelligent user interface
ENVIRONMENTAL POLLUTION EXPERT
2nd Conference on AI Applications
USA
PROGRAMMING
1985

ESCOR
ENGINEERING
real-time process control system
ESCORT
Computing (mag) 28/11/85 p13
developed by PACTEL and commercially available on Xerox 1100 machines UK
CONTROL
1985

EURIS
VARIOUS
works in a similar fashion to AM but in non-mathematical subjects
EURISKO
Goodall (1985) p65
written by Lenat as a follow-on to AM. USA
PREDICTION
1982

FEASA
COMPUTER-Software
advises engineers using a highly complex program called NASTRAN
FINITE ELEMENT ANALYSIS SPECIFICATION AID
Expert Systems User May 1985 p14
developed by Ian Taig of British Airways UK using Savior as a front-end to a CAD package
ADVISOR
1985

GEOLO
COMPUTER-Software
an expert system front end for MENDEL
GEOLOG
Hewett, Timms & d'Aumale (1986)
developed by Shell NETHERLANDS
PROGRAMMING
1986

GLIM
COMPUTER-Software
an intelligent front-end to a statistics package
GLIM
Hewett, Timms & d'Aumale (1986)
developed at Imperial College and NAG UK
Alvey IKBS project 033
PROGRAMMING
1986

GPLAN
COMPUTER-Software
system interrogates large databases to satisfy queries and plans out the route through the database
GPLAN
Smart and Langeland-Knudsen (1986) p101
developed at Purdue University USA
PROGRAMMING
1975

GPSI
COMPUTER-Software
system debugs certain compile time and run time errors in Pascal programs. It has been found that there is also a tutoring application for students having their routines debugged.
GPSI
Smart and Langeland-Knudsen (1986) p101
written in Pascal for IBM PCs at The University of Illinois USA
DEBUGGING
1983

GPS
a planning program which attempted to generate a plan to achieve a goal state
GENERAL PROBLEM SOLVER
Newell and Simon (1963)

system was not successful on large complex
problems. USA
PLANNING
1959

GREAT
COMPUTER-Software
teaches students to write programs in LISP
Goal-Restricted Environment for Tutoring And
Educational Research System
BYTE April 1985 p159
implemented in FranzLisp on VAX computers at
Carnegie-Mellon University USA
TUTOR
1985

GUIDO
MEDICINE
At its simplest, it is MYCIN rearranged for
tutorial purposes
GUIDON
Barr and Feigenbaum (1982)
developed at Stanford University as part of
the MYCIN project USA
TUTOR
1979

HEXSC
MILITARY
an experimental system designed to deal with
control problems encountered in military and
advanced industrial applications
HEXSCON
IEEE Software March 1986 p16
designed to hold up to 5000 rules in a 512K
micro USA
CONTROL
1986

HIFSQ
MEDICINE
system acts as a diagnostic aid in
opthamology. May be developed for use in
teaching strabismus diagnosis
HINT FOR SQUINT
Smart and Langeland-Knudsen (1986) p113
developed at John Hopkins University USA
using FORTRAN
DIAGNOSIS
1983

IIDA
COMPUTER-Software
Individualised Instruction for Data Access.
System aids the searching of online
scientific and bibliographic information
retrieval systems
IIDA
Smart and Langeland-Knudsen (1986) p119
developed at Drexel University USA using
MIT's Multics computer via TELENET
PROGRAMMING
1982

KARDI
MEDICINE
a system for the electrocardiographic
diagnosis of cardiac arrhythmias
KARDIO-E
Expert Systems vol 2-1 Jan 1985 p46
written in YUGOSLAVIA using Prolog with a
knowledge base of over 8300 rules
DIAGNOSIS
1985

KBMS
COMPUTER-Software
system aids in the intelligent retrieval of
information from very large databases
KNOWLEDGE BASE MANAGEMENT SYSTEM
Smart and Langeland-Knudsen (1986) p132
developed at Stanford University USA
PROGRAMMING
1980

KIWI
COMPUTER-Software
system is planned to allow the user to
manipulate a number of databases through an
integrated knowledge based interface
KIWI
Smart and Langeland-Knudsen (1986) p134
being developed using PROLOG as an ESPRIT
project (Italy, France, Belgium, Denmark and
Netherlands)
ADVISOR
1985

KM1
COMPUTER-Software
system plans and executes the strategies for deductive database search and query answering
KNOWLEDGE MANAGEMENT 1
Smart and Langeland-Knudsen (1986) p134
implemented on a Xerox 1100 LISP machine by System Development Corporation, USA
PROGRAMMING
1983

KNOES
COMPUTER-Software
the project is an attempt to build an expert system whose knowledge is encyclopaedic and the retrieval of that information is aided by the system
KNOESPHERE
Smart and Langeland-Knudsen (1986) p136
system is planned to be written in LISP using Symbolics and Xerox LISP machines in the USA
PROGRAMMING
1983

LADBR
BUSINESS
system aims to teach betting shop branch managers the implications of various rules used in setting odds
LADBROKES
Expert Systems User Jan 1987
developed by ITDU at Kingston College of F.E. UK for Ladbrokes Ltd
TUTOR
1986

LINK
MANUFACTURING
a control system for a cement kiln to optimise the use of fuel
LINKMAN
Expert Systems vol 2-2 April 1985 p88
developed by SIRA (UK) and Blue Circle Cement Co. on a DEC PDP/11
CONTROL
1985

LRS
LAW
information retrieval system for Negotiable

Instruments Law
LEGAL RESEARCH SYSTEM
Smart and Langeland-Knudsen (1986) p149
developed at the University of Michigan USA
PROGRAMMING
1981

MACSY
MATHEMATICS
a large interactive system designed to assist in solving mathematical problems
MACSYMA
Barr and Feigenbaum (1982) p143
developed by Moses et al using LISP at MIT USA
ADVISOR
1968

MEDIC
MEDICINE
Designed to diagnose the cause of acute abdominal pain
MEDICL
Computer Weekly 21/8/86 p56
Marketed by ICL UK
DIAGNOSIS
1985

MENO
COMPUTER-Software
system is designed to help novice programmers learn Pascal
MENO
Smart and Langeland-Knudsen (1986) p164
written in LISP and Pascal at the University of Massachusetts and Yale USA
TUTOR
1981

MIRIA
COMPUTER-Software
an intelligent front-end to EDF's enormous staff database
MIRIAM
Expert Systems User March 1987
developed by EDF FRANCE
PROGRAMMING
1986

MOLGE
SCIENCE
assists in the design of genetic experiments
MOLGEN
Alty and Coombs (1984) p161
developed at Stanford University USA using
KEE on a Xerox 1108
DESIGN
1979

MYCIN
MEDICINE
provides consultative advice on the diagnosis
and therapy for infectious diseases
MYCIN
AI Handbook vol 2 p184
written in Lisp by Shortliffe et al at
Stanford University USA
DIAGNOSIS
1976

NAVEX
AEROSPACE
an experimental system designed to perform
the task of a NASA mission control operator
and help land the Space Shuttle. It was never
used in practice, but it outshone human
operators in tests
NAVIGATION EXPERT SYSTEM
Expert Systems User April 1985 p12
developed at NASA using Inference ART USA
CONTROL
1985

NEOMY
MEDICINE
an infectious disease consultant used for
teaching by GUIDON
NEOMYCIN
IJMMS 20, 1984 p3
developed at Stanford USA as part of the
MYCIN project
TUTOR
1981

NTGAS
COMPUTER-Software
a front end to ICL's Querymaster software
allowing users with little or no knowledge of
the query language to access data held in the

database
NORTH THAMES GAS
Hewett, Timms & d'Aumale (1986)
developed by North Thames Gas UK using ICL's Adviser shell
PROGRAMMING
1986

PFES
MANUFACTURING
system aims to design an optimal mixture of ingredients from a large selection to meet both chemical and physical constraints and commercial factors
PRODUCT FORMULATION EXPERT SYSTEM
Hewett, Timms & d'Aumale (1986)
Alvey IKBS project 052 developed by Shell Research UK
DESIGN
1986

PHOTO
COMPUTER-Hardware
system diagnoses faults and recommends treatment during the manufacturing process of integrated circuit wafers
PHOTOLITHOGRAPHY ADVISOR
SIGART Newsletter 92 April 1985 p42
developed on Hewlett Packard 9000 Series 200 workstations in LISP at HP Research Labs USA
DIAGNOSIS
1985

PIA
BUSINESS
system helps researchers analyse the performance and pinpoint the problems of Regional Health Authorities. Data is not keyed in, but is picked up from ASCII datafiles
PERFORMANCE INDICATOR ANALYST
Expert Systems in Business vol 1-1
developed by Coopers and Lybrand, using Crystal on an IBM PC, for the Operational Research Service of the DHSS UK
ADVISOR
1987

PICON
MANUFACTURING
provides advice to process operators
PROCESS INTELLIGENT CONTROL
Expert Systems User Jan 1986 p18
Runs on Lisp Machines Lambda computers and is commercially available USA
CONTROL
1984

PROSP
GEOLOGY
assists geologists prospecting for mineral deposits
PROSPECTOR
Barr and Feigenbaum (1982) p155
developed by Duda using INTERLISP USA. The knowledge-base is kept separate from the mechanisms that use the knowledge
PREDICTION
1978

PROUS
COMPUTER-Software
analyses and aids the debugging of Pascal programs
PROGRAM UNDERSTANDER FOR STUDENTS
BYTE April 1985 p106
written in LISP on a VAX 11/750 at Yale University USA. A smaller version runs on an IBM PC
TUTOR
1984

PUBLI
TRAVEL
assists in the handling of public transport queries. Effectively acting as a front end to a database
PUBLIC TRANSPORT QUERIES SYSTEM
Expert Systems User Jan 1986 p7
developed by Software Science and a consortium of travel companies as part of the Alvey TRACE club UK
PROGRAMMING
1986

PUFF
MEDICINE
diagnoses pulmonary function disorders using

data from respiratory tests
PULMONARY FUNCTION PROGRAM
AI Handbook vol 2 p180
developed by Kunz et al containing 55 rules
at Stanford University USA
DIAGNOSIS
1978

QUICK
MEDICINE
as well as its advisory role, it can also be
used as a teaching aid
QUICK INDEX TO CADUCEUS KNOWLEDGE
Smart and Langeland-Knudsen (1986) p225
written in C on a VAX 11/780 under UNIX at
the University of Pittsburgh USA
ADVISOR
1985

RESED
COMPUTER-Software
an intelligent information retrieval system
for data relating to French historical
information
RESEDA
IJMMS 20 1984 p87
developed at the National Centre for
Scientific Research, Paris FRANCE
PROGRAMMING
1977

SACON
COMPUTER-Software
system advises nonexpert engineers in the use
of a general purpose computer program for
structural analysis (MARC)
STRUCTURAL ANALYSIS CONSULTANT
Smart and Langeland-Knudsen (1986) p241
developed using EMYCIN at Stanford University
USA (150 rules)
ADVISOR
1979

SCHOL
EDUCATION
a dialogue system designed to review a
student's knowledge of South America
SCHOLAR
Carbonell (1970)

the data is represented as a semantic network
USA
TUTOR
1970

SECOF
ENGINEERING
system is used as a training tool to advise
on drill-bit sticking problems in oil wells
SECOFOR DRILLING ADVISOR
Jones & Davies (1986) p105
developed for Elf-Acquitaine USA
ADVISOR
1983

SHUT1
AEROSPACE
system acts as a real time navigation
assistant to mission control personnel for
high speed Shuttle re-entry
SHUTTLE 1
Hewett and Sasson (1986) p173
developed by NASA USA
MONITORING
1986

SMART
COMPUTER-Software
an intelligent front end to SQL/DS, intended
to ease access to the databases by the casual
user
SMARTY
Hewett, Timms & d'Aumale (1986) developed by
IBM FRANCE using ESE
PROGRAMMING
1986

SOPHI
EDUCATION
a tutor and natural language processor
applied to the field of electronics
SOPHISTICATED INSTRUCTION ENVIRONMENT
IJMMS 7 1975 p675
developed at Bolt, Beranek and Newman USA in
LISP on a DEC PDP 10
TUTOR
1975

SOUP
MANUFACTURING
a fault diagnosis system for soup cookers
SOUP
Hewett and Sasson (1986) p134
developed by Texas Instruments and Campbell's Soup USA
DIAGNOSIS
1985

SPECT
ENGINEERING
it enables a triple mass spectrometer to be tuned dynamically when in use, whereas before it had to be taken out of use for tuning
SPECTROMETER TUNER
Expert Systems User Nov 1985 p20
built using KEE on a Xerox Lisp machine at a USA government laboratory
REPAIR
1985

SPHIN
MEDICINE
a diagnosis system for epigastric pains and jaundice, also used as a teaching aid
SPHINX
Smart and Langeland-Knudsen (1986) p254
developed at the Universite Marseille, FRANCE with a 400-rule knowledge base
DIAGNOSIS
1983

SUS
MILITARY
monitors the many different kinds of information available to the commander of a naval vessel
SIGNAL UNDERSTANDING SYSTEM
Goodall (1985) p63
developed by SPL and the Admiralty Research Establishment UK
MONITORING
1983

TADIS
FINANCE
advises dealers on fluctuating markets. In a three week trial period it is reported that it 'played' the foreign exchange market and

performed better than City institutions
TADIS
Datalink 14/4/86
written in C for DEC VAX machines by Data
Logic UK
ADVISOR
1986

TEIRE
COMPUTER-Software
system assists in entering and updating
knowledge bases such as MYCIN
TEIRESIAS
Barr and Feigenbaum (1982) p87
developed at Stanford University by Davis
using INTERLISP USA
PROGRAMMING
1976

THOMD
MEDICINE
system provides advice for the treatment of
diabetes
THOMAS DIABETES
Smart and Langeland-Knudsen (1986) p271
developed at St Thomas's Hospital Medical
School and City University UK
ADVISOR
1985

WELDS
COMPUTER-Software
an intelligent front end to a database of
metal types, welding processes and related
information. It produces an ordered list of
the best electrodes for use on the specified
job
WELD SELECTOR
Expert Systems User June 1986
developed by the Colorado School of Mines
and the American Welding Institute USA on an
IBM PC AT
PROGRAMMING
1986

WHEAT
AGRICULTURE
advises on the use of fungicide on wheat
crops

WHEAT COUNSELLOR
Computing (mag) 5/12/85 p14
part of the ICI Counsellor range which is
commercially available UK
ADVISOR
1984

XCON
COMPUTER-Hardware
a configuration system for VAX and PDP
machines
EXPERT CONFIGURER
Computer Weekly 10/7/86 p10
developed by DEC USA with a rule base of over
2000 rules
PLANNING
1980

XSEL
COMPUTER-Hardware
developed as a front end for R1, system
assists DEC salesmen to select components
that satisfy the customer's application
XSEL
Machine Intelligence 10 1982 p325
developed by DEC and Carnegie-Mellon
University USA
PROGRAMMING
1982

YES/M
COMPUTER-Hardware
a continuous real-time system that exerts
active control over the MVS operating system
YORKTOWN EXPERT SYSTEM/MVS MANAGER
IBM Journal of R&D vol 30-1 Jan 1986 p14
developed with OPS5 (500 rules), USA
CONTROL
1985

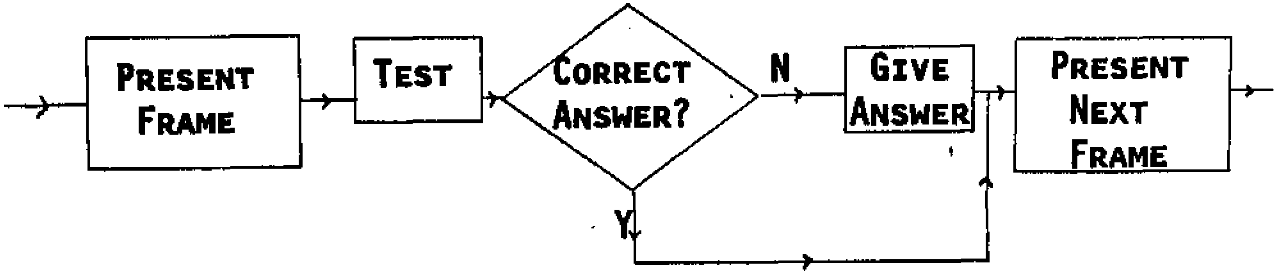
APPENDIX 5

Computer-based Training

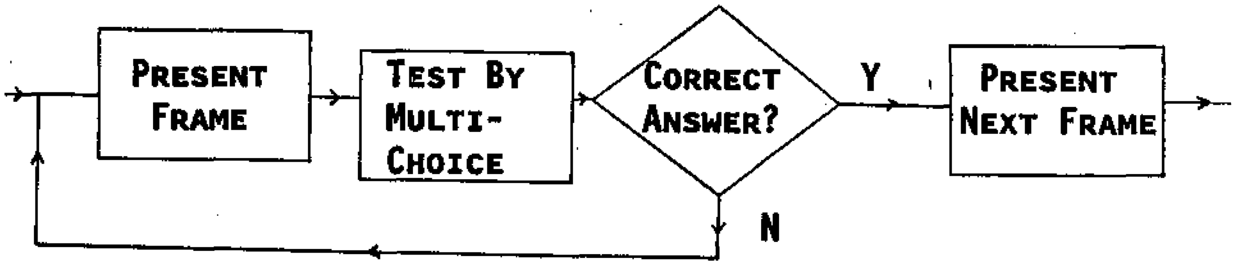
Flowcharts

LINEAR PROGRAMS

SKINNER'S PROGRAM

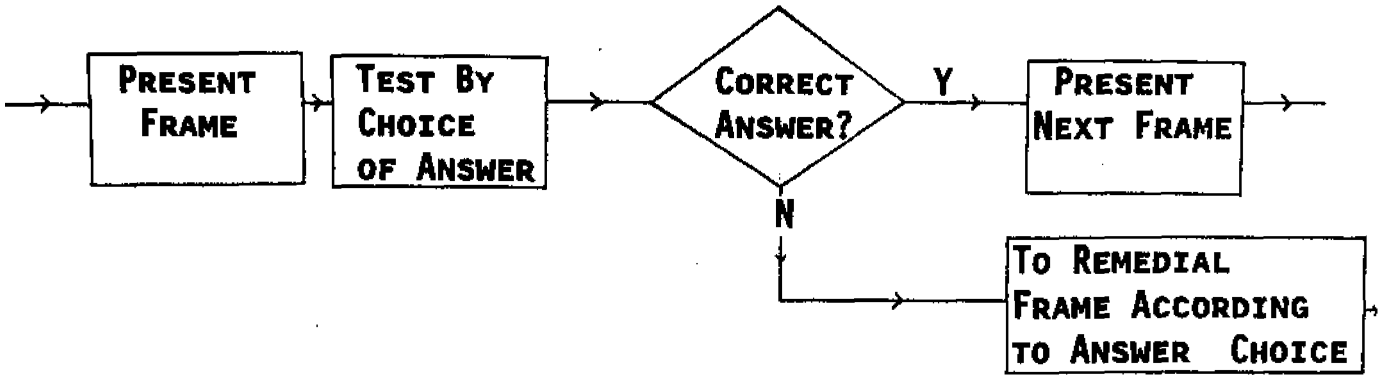


PRESSEY'S PROGRAM

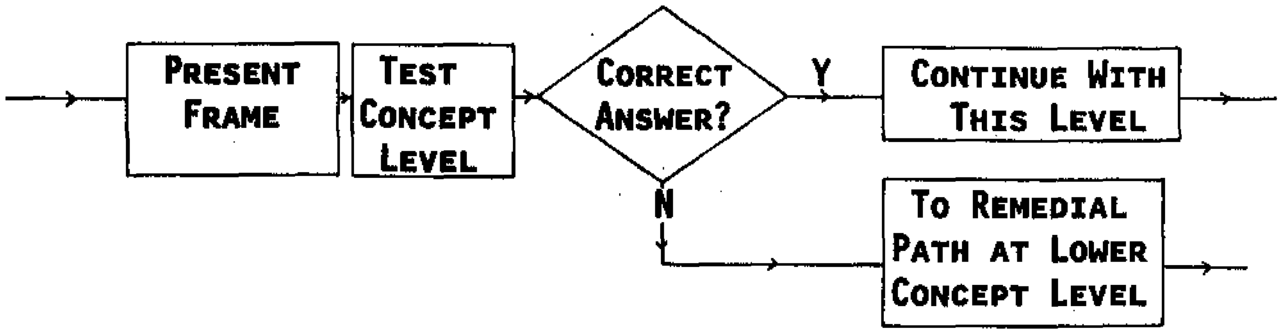


BRANCHING PROGRAMS

CROWDER'S PROGRAM



KAY'S PROGRAM



APPENDIX 6

NCC Survey

Questionnaires

NCC THE NATIONAL CENTRE FOR INFORMATION TECHNOLOGY

The National Computing Centre Limited Oxford Road Manchester M1 7ED
Telephone 061-228 6333 Telex 668962
*Telecom Gold 81:NCT 008; Fax (Groups 2 & 3): 061-228 2579; Prestel * 600182 #*

22nd August 1986

EXPERT SYSTEMS AND EDUCATION

Dear Colleague,

I am a teacher researching the use and potential of Expert Systems and Artificial Intelligence (A.I.) within the educational environment (in its widest context). This research will form part of the basis for a Master's Degree thesis.

In order to establish the present and future scope and direction of such activity, I am carrying out, in association with the National Computing Centre Ltd., a survey on the use of the Alvey/NCC Expert Systems Starter Pack and other products. I would be grateful if you could spare some of your time to complete the enclosed questionnaire. Space has been left for you to make additional comments, criticisms and suggestions. Please add any extra information that you feel will be interesting, relevant or useful.

If this request has landed on the wrong desk, could you please pass it on to the appropriate person. On completion return it to me in the enclosed reply envelope, which uses our FREEPOST address.

Direct any other communications to NCC at the above address marked for the attention of John Heslop or Mike Newman, both of whom are Senior Consultants in the Knowledge Management Systems Group.

A summary of the report will be available, on request, to all respondents.

Thank you for your time and trouble.

Yours sincerely,

Section 1 - This section asks questions relating specifically to the Alvey/NCC Expert Systems Starter Pack.

1.1 Please give a brief summary of the reasons for purchasing the Starter Pack.

1.2 Please indicate by a "x" in the relevant boxes, your impressions of the pack and its contents.

	Not used	Poor	Adequate	Good	Excellent
Software:					
Micro Expert	[]				
Ease of use		[]	[]	[]	[]
Relevance		[]	[]	[]	[]
Micro Synics	[]				
Ease of use		[]	[]	[]	[]
Relevance		[]	[]	[]	[]
Expert Ease	[]				
Ease of use		[]	[]	[]	[]
Relevance		[]	[]	[]	[]
ESP Advisor	[]				
Ease of use		[]	[]	[]	[]
Relevance		[]	[]	[]	[]
Documentation:					
Ease of use		[]	[]	[]	[]
Relevance		[]	[]	[]	[]
Complete pack:					
Value for money		[]	[]	[]	[]
Usefulness		[]	[]	[]	[]

1.3 How much use has been made of the pack ?

Number of users

Only one	[]
Less than 5	[]
5 - 10	[]
Over 10	[]

Frequency of use

Every day	[]
Every week	[]
Every month	[]
Less frequently	[]

1.4 Who has used the pack ?

Teaching staff [ ]
Non teaching research staff [ ]
Students [ ]
Others - please specify

1.5 Please indicate by a "x" in the relevant boxes, details of the uses of the pack.

Demonstrations [ ]
Seminars [ ]
Teaching [ ]
As the basis of a course [ ]
As part of a course [ ]
Awareness of Expert Systems [ ]
Evaluation of Expert Systems [ ]
Familiarisation with E.S. [ ]
Research [ ]
Prototyping [ ]
Development of applications [ ]
Administrative uses [ ]
Others - please specify

1.6 Please give details of the planned use of the pack during the next twelve months.

1.7 Please give any other comments you have to make about the pack.

Section 2 - This section contains questions relating to other Expert Systems/A.I. software products.

2.1 Please list any other Expert Systems/A.I. software products that you use e.g. shells, languages, authoring systems.

2.2 Please give details of the use of these products.

2.3 Please indicate any areas where you have developed, or are developing, Expert Systems.

Mathematics	[]	Chemistry	[]	Medicine	[]
Computer Studies	[]	Geology	[]	Accountancy	[]
Computer Science	[]	Education	[]	C.A.D.	[]
Business Management	[]	Data Processing	[]		
Mechanical Engineering	[]	Electrical Engineering	[]		
Chemical Engineering	[]	Production Engineering	[]		
Others - please specify					

Please give details of, and/or references to, these applications and stating whether each application was a demonstration, prototype or full system. (continue overleaf if necessary)

2.4 Please list any Expert Systems/A.I. software products that you are planning to purchase within the next twelve months.

Section 3 - This section contains questions relating to general interest in Expert Systems/A.I. and your view of the future of this technology.

3.1 Is there an interest in Expert Systems in any other department of your institution ?
If yes please give details.

3.2 Is there an interest within your institution in Intelligent Teaching Systems (Intelligent CBT, CAI etc.) ?
If yes, please give further details.

3.3 Please give your assessment of the future role of Expert Systems within education.

3.4 Any other comments.

NCC THE
NATIONAL CENTRE
FOR INFORMATION
TECHNOLOGY

The National Computing Centre Limited Oxford Road Manchester M1 7ED
Telephone 061-228 6333 Telex 668962
Telecom Gold 81:NCT008; Fax (Groups 2 & 3): 061-228 2579;

20th August 1987

EXPERT SYSTEMS AND EDUCATION

Dear Colleague,

You recently received a copy of my report entitled 'The use of the Alvey/National Computing Centre Expert Systems Starter Pack within Higher and Further Education'. I hope that you found it interesting and that it proved useful to you in your work.

I am continuing my research, which, as you may recall, will form part of the basis for a Master's Degree thesis. I would like to continue to monitor and analyse developments, not only with the NCC Starter Pack, but with other A.I./Expert System products. Hence I would be grateful if you could spare some of your time to complete the enclosed questionnaire. Space has been left for you to make additional comments and any extra views or information that you can give would be welcomed.

If this request has landed on the wrong desk, could you please pass it on to the appropriate person. On completion return it to me in the enclosed reply-paid envelope which uses the NCC FREEPOST address. A summary report will again be available, on request.

Thank you again for your time and trouble.

Yours sincerely,

Section 1 - This section asks questions relating specifically to the Alvey/NCC Expert Systems Starter Pack.

1.1 How much use has been made of the Pack during the period August 1986 - August 1987 ?

Number of users		Frequency of use	
Not used	[]		
Only one	[]	Every day	[]
Less than 5	[]	Every week	[]
5 - 10	[]	Every month	[]
Over 10	[]	Less frequently	[]

1.2 Who has used the Pack ?

Teaching staff []
Non teaching research staff []
Postgraduate students []
Undergraduate students []
Others - please specify

1.3 Please indicate by a 'x' in the relevant boxes, details of the uses of the Pack.

Demonstrations []
Seminars []
Teaching []
As the basis of a course []
As part of a course []
Awareness of Expert Systems []
Evaluation of Expert Systems []
Familiarisation with E.S. []
Research []
Prototyping []
Development of applications []
Administrative uses []
Others - please specify

1.4 Please give details of the planned use of the Pack during the period August 1987 - August 1988.

1.5 Please give any others comments you have to make about the Pack.

Section 2 - This section asks questions relating to other Expert Systems/A.I. products.

2.1 Please list any Expert Systems/A.I. software products that you use e.g. shells, languages, authoring systems.

2.2 Please list the hardware that you use in conjunction with the products listed in (2.1).

2.3 Please list any hardware or software products that you are planning to purchase in the next twelve months.

Section 3 - This section asks questions relating to curriculum areas and applications.

3.1 Please indicate any curriculum areas where either the Pack or the software mentioned in (2.1) have been used.

Curriculum area	Software used

3.2 Please use the blank forms provided to give details of any applications that have been developed. Failed systems do not receive much publicity, but there is often much to learn from 'failures' and I would also like to hear of systems that did not 'succeed' and your appraisal of the reasons for 'failure'.

Application title
Curriculum area
Software and hardware used
Scope of system (e.g. Demo Prototype Full)
Size of system (e.g. number of rules)
Number of man-days to develop
Author (e.g. Lecturer Postgrad. Undergrad.)
Reference (if applicable)
Brief outline of what the system does (or doesn't) do

Please photocopy and distribute the forms to any other people who are developing Expert Systems within your Institution.

Section 4 - This section asks questions relating to general interest in Expert Systems/A.I.

4.1 How many people in your institution have, or will have, an interest in, or are developing, Expert Systems ?

Number of people	Aug 86	Aug 87	Aug 88
An interest in E.S.	......	......	......
Developing E.S.	......	......	......

Please give details

4.2 Please compare the present level of interest within your institution in Intelligent Teaching Systems (Intelligent CBT, CAI etc.) with the position in August 1986.

Less than at August 1986 [ ]
About the same [ ]
Increased interest [ ]
Please give details

4.3 Any other comments

GLOSSARY OF SOME OF THE TERMS, AND ABBREVIATIONS USED IN THIS PAPER

I am grateful to Forsyth and Rada (1986) for the following definition of 'glossary'

> "a list of mystifying definitions not containing the word you seek".

I am further encouraged by Zigler (1986)

> "a definition cannot be right or wrong, only more or less useful"

I trust that this glossary is sufficiently detailed without being too large.

Adaptive control system - a system using feedback to adjust parameters controlling the action of the system and maintaining optimal performance under changing conditions

AI - artificial intelligence

Algorithm - a precise description of how to solve some specific problem, which will work (although it may take a long time)

APES - Augmented Prolog for Expert Systems

Architecture - the structure of the computer circuits

Artificial intelligence - a branch of computer science involved with making computers solve problems in an intelligent fashion

Attribute - a variable or single-argument used in asserting one property of an object or situation

Backtracking - retracing the latest step in the search for a solution when it has led to a dead end

Backward chaining - a system of reasoning backwards from hypotheses to the evidence needed to support or refute those hypotheses

Bayes rule - a theorem, widely used in expert systems, concerning conditional probabilities

BCS - British Computer Society

Belief - a measure of the degree of truth (as in certainty factor)

Blackboard - a data base accessible to independent knowledge sources and used by them to communicate with one another

Blame assignment - identification of the rules or decisions responsible for failure in reaching a goal

Boolean algebra - a system where something is true (1) or false (0)

Branching program - a teaching program organised as a set of frames, the route through the frames being dependent upon the user's responses

Breadth-first search - a technique of search where all the potential solutions at a particular level are considered before going on to the next level down

Bug - an error in a computer program

C - a low level general purpose programming language

CAL - Computer aided (or assisted) learning. Note that I have used the term 'CAL' as a general acronym to cover all uses of Computers AND Learning, rather than individually specifying 'CAI' (computer assisted instruction), 'CML' (computer managed learning), 'CBT' (computer based training) and 'CBL' (computer based learning) etc.

Certainty factors - numerical weightings used to estimate a conclusion's degree of truth

CMU - Carnegie-Mellon University

Conclusion - the part of a rule which is to be executed after its conditions are found to be true

Conditional probability - the conditional probability of X given Y is the probability that X occurs given that we know Y occurs

Conflict resolution - the technique of resolving the problem of multiple matches in a rule-based system

Credit assignment - identification of the rules or decisions responsible for success in reaching a goal

Critic - the component of a learning system that evaluates proposed rules or concepts and carries out credit assignment

Courseware - computer training material. Software and accompanying documentation that forms the basis for CAL

DARPA - Defence Advanced Research Projects Agency (A department of the US Defence Department)

Data structure - the organised form in which grouped data items are held in the computer

DBMS - Database management system

Debugging - the activity of finding and removing bugs in programs

DEC - Digital Equipment Corporation

Decision support system - a computer system that provides information to assist the human decision-maker

Decision tree - a series of tests, arranged in a tree-like fashion, that lead on to other tests. By applying the tests and following the appropriate branches you eventually arrive at the correct place

Declarative knowledge - forms of knowledge that make assertions about entities and the relationships between them

Declarative programming - the technique of giving the computer a description of a situation and a goal and letting it work out the solution through logic rather than by specifying the steps to be taken (as in procedural programming)

Default value - a predetermined value which the system will assume is true unless told otherwise

Demon - a self contained part of a computer program that will be triggered by a particular predetermined set of circumstances

Depth-first search - a search strategy which follows a single branch of the search tree until it arrives at the solution or a dead end

DES - Department of Education and Science

Dialogue - a sequence of messages between the user and the computer

Domain - a subject area

DSS - Decision support system

Dynamic memory - memory organisation that continually readjusts itself to conform to new facts

End-user - the person for whom the system was developed

ES - expert system(s)

Exhaustive search - the method of trying all possible solutions, in a brute force fashion, in the hope of finding an acceptable one

Expert system - a computer program that uses expert knowledge to attain high levels of performance in a narrow problem area

Expert system building tool - the programming language and support package used to build the expert system

Explanation facility - part of an expert system that explains how it arrived at a decision and justifies the steps taken to reach it

FEU - Further Education Unit

FGCS - Fifth Generation Computing Systems

Field - one aspect of a record in a database

Formal logic - a study of correct reasoning which has been mechanised in the language PROLOG

Forward chaining - a method of reasoning from evidence to conclusions

Frame - a data structure which describes an object. It has a number of slots which are filled with attributes

Fuzzy logic - a theory, based on Boolean algebra, founded by Zadeh in the early 1960s, in which each proposition has a fractional degree of truth

Generalisation - extending the scope of a concept or rule to cover more examples

Generative program - a teaching program which produces questions and text from partial specifications while it is running

Goal - the current objective or sought conclusion

Granularity - the level of detail in a piece of information

HCI - Human computer interaction

Heuristic method - problem solving based on rules of thumb (heuristics), which work by trial and error

Hypertext - A system of interactive linked text that allows 'jumps' to be made between text items rather than the conventional sequential fashion. (The July 1988 edition of the Communications of the ACM was a special issue about Hypertext)

IBM - International Business Machines

ICAI - Intelligent Computer Assisted Instruction

IFE - Intelligent Front End (a user-friendly interface to a software package)

IJCAI - International Joint Conference on AI

IJMMS - International Journal of Man-Machine Studies

IKBS - Intelligent Knowledge Based System

Induction - the process of deriving general rules from particular examples

Inference engine - the part of an expert system that draws new conclusions from given facts

Inference network - a diagrammatic form of knowledge representation involving statements and rules, showing how the truth of one assertion influences the truth of others

Information retrieval - the process of matching a query against a database and selecting those items which are relevant to the query

Inheritance hierarchy - a knowledge representation in which items are held in a tree structure. Items at lower levels of the tree inherit properties belonging to their ancestors higher up the tree

IR - Information Retrieval

IT - Information Technology

ITDU - Information Technology Development Unit (based at Kingston College)

JCAL - Journal of Computer Assisted Learning

KBDSS - Knowledge-based decision support system

KBS - Knowledge Based System

Knowledge acquisition - the process of obtaining knowledge

Knowledge base (kb) - the database of an expert system

Knowledge engineer - a person who obtains knowledge from human specialists and encodes it in the computer

Knowledge representation - the choice of data structures to represent information in the computer

Learning programs - programs which are designed to improve their performance as a result of feedback

Linear program - a teaching program organised as a set of frames, each of which always goes on to the same next frame in a serial fashion

LISP - a list processing language

Logic programming - an attempt to transform computers from calculating engines to inference engines, often using PROLOG

LSI - Large Scale Integration

LTM - Long Term Memory

Machine induction - the process whereby a computer program learns rules or concepts from examples presented to it

Machine learning system - a system which improves its performance by amending the knowledge base using inductive methods

Meta-rule - a rule that describes how other rules should be used

MIS - Management Information System

MIT - Massachusetts Institute of Technology

MMI - Man-Machine Interface

MOP - Memory Organisation Packet (Schank 1972)

MSC - Manpower Services Commission

NCC - National Computing Centre

NCET - National Council for Educational Technology

Node - a junction in a tree structure

Noise - distorted information

Object-oriented programming - a method of programming in which the elements are objects, arranged in an inheritance hierarchy, which communicate by passing messages

OLS - Open Learning System

Parallel processing - a fast and powerful method of computer processing where many instructions can be processed simultaneously

Parameter adjustment - a primitive form of incremental learning where the relative weightings of coefficients in a mathematical expression are adjusted as a result of feedback

Parser - a system used to decompose a sentence into its grammatical components

Pattern recognition - a data-reduction task in which the system takes input data and assigns it to one of two or more classes

PC - Personal computer

Phoneme - a unit of significant sound in a word

Predicate calculus - a form of symbolic logic where propositions are composed of predicates and relations between objects linked by AND and OR operators

Procedural programming - the conventional way of driving a computer by giving it a sequence of instructions

Procedural knowledge - knowledge describing what to do with facts

Production rule - a rule which will carry out an associated action if its conditions are satisfied

PROLOG - a programming language based on logic

Pruning - reducing the search space by narrowing the alternatives

Query language - a set of rules governing the formulation of questions for searching a database

Real time system - a system which operates in 'real time' and responds to situations as they occur

RML - Research Machines Ltd

Robustness - the quality of a system

particularly when it is pushed to its limits or given incomplete or inconsistent data

Rote learning - learning by the storage of facts, without generalisation

Rule base - synonymous with knowledge base

Rule-based system - a program which operates on the basis of production rules

Search space - an abstract space of all potential solutions

Search tree - programs which try to search intelligently for solutions produce tree-like structures which branch out as the various options and their consequences at each stage are considered

Semantic network - a knowledge representation scheme in which nodes stand for objects and arcs linking nodes stand for the relations between those objects

Serial processing - the traditional method of computer processing where a single instruction is processed at a time

Shell - a software environment with the application of building an expert system

Signature table - a table in which

combinations of features are used to index or address information concerning the state represented by that combination of features

Slot - a field in a frame

Specialisation - the narrowing of the scope of a rule so that it covers fewer examples

STM - Short Term Memory

Symbolic reasoning - problem solving based on the application of strategies and heuristics to manipulate symbols standing for problem concepts

TAP - Training Access Point

Toy problem - an artificial problem, such as a game, or an unrealistic adaptation of a complex problem

Training set - the set of examples given to a learning system to enable it to induce new knowledge

Transputer - a powerful parallel processor produced on a single chip by INMOS

ULSI - Ultra Large Scale Integration

Uncertainty - the measure of how much confidence is placed in a piece of knowledge used in uncertain reasoning

Value - the relevance of a statement depends not only upon the level of belief, but also on the risk or value associated with it. A condition may be highly important even if it is unlikely

VLSI - Very Large Scale Integration - the system of producing more components onto single chips of silicon

Working memory - the memory area in a production system used for short term information (intermediate calculations, messages passed between rules)

REFERENCES AND BIBLIOGRAPHY

The following list of publications have been used during the research for this thesis. Some have been used for general information and background reading, while others have been specifically referred to in the text. Other texts have been listed to act as signposts, for the reader, to sources of further information and analysis.

Accountant's Magazine (1985) Expert Systems - threat or opportunity ? The Accountant's Magazine December 85 p546

Adam A E and Taylor A D (1986) Modelling analogical reasoning for legal applications IN Bramer (1987)

Adams D (1978) A hitchhikers guide to the galaxy Fontana

Addis T R (1982) Expert systems: an evolution in information retrieval' Information Technology: Research and Development 1 p301-324

Addis T R (1985) Designing knowledge-based systems Kogan Page

Aitkenhead A M and Slack J M (1985) Issues in cognitive modelling Lawrence Erlbaum Associates

Aleksander I (1984) Myths that are spoiling Britain's IT chances Guardian 12/9/84

Alexander I (1987) Explanation in Expert Systems Computing 16/4/87 p29

Allan B (1984) Computerised Information Retrieval Systems for Open Learning IN van Rijsbergen (1984)

Allen B (1984) Introducing LOGO Granada Publishing

Alty J (1985a) Expert Systems NCC/Alvey Starter Pack IN NCC (1985)

Alty J (1985b) Journal of Computer Aided Engineering February 1985

Alty J and Coombs M J (1984) Expert systems; concepts and examples NCC Publications

Alvey Committee (1982) A programme for advanced IT HMSO

Alvey Mailshot (1987) Towards an intelligent help file finder Alvey IKBS project 017 Alvey Mailshot 9/87

Alvey P L and Greaves M F (1986) Observations on the development of a high performance system for leukaemia diagnosis IN Bramer (1987)

Anderson J R (1982) The acquisition of cognitive skill Psychological Review vol 89-4 p369-406

Anderson J R (1983) Knowledge compilation: the general learning mechanism IN Michalski (1983)

Anderson J R and Reiser B J (1985) The Lisp Tutor Byte 10 p159-175

Anderson J S A (1985) The teacher and the researcher - who pays the ferryman ? IN Smith (1985)

Andree R V (1958) Programming the IBM650 magnetic drum computer and data processing machine Holt

Aronofsky A (1969) Progress in Operations Research vol 3 John Wiley

Arthur C (1986) Computer Weekly 2/10/86 p28

Arthur C and Watts S (1986) Doctors diagnose legal snag Computer Weekly 21/8/86 p56

Asimov I (1967) I, Robot Granada Publishing

Atkinson R C (1976) Adaptive instructional systems: some attempts to optimise the learning process IN Klahr (1976)

Ausubel D P (1968) Educational Psychology: a cognitive view Holt, Reinhart and Winston

Bachtin O (1984) It is what it's used for: job perception and system evaluation Interact '84 First IFIP Conference on Human Computer Interaction

Bailey R W (1982) Human performance engineering: a guide for systems designers Prentice Hall

Bainbridge B G (1986) Using Xi in 6th form computer studies IN Thorne (1986)

Bainbridge L (1981) Verbal reports as evidence of the process operator's knowledge IN Mamdani and Gaines (1981)

Baker A (1984) Speculations on the future of knowledge engineering in Europe 1 IN Bernold and Albers (1984) p181-183

Bandler W and Kohutt L J (1980) Semantics of implication operators and fuzzy relational products IJMMS vol 12 1980 p89-116

Banks M (1986) DEC feels a market pull Computer Weekly 3/4/86 p16

Barchan J, Woodmansee B J and Yazdani M (1985) A Prolog based tool for French grammar analysers Instructional Science vol 14

Barfield W (1986) Expert-novice differences for software: implications for problem solving and knowledge acquisition Behaviour and Information Technology vol 5-1 p15-29

Barker P (1987) Knowledge based CAL Computer Education February 1987 p4-7

Barr A and Feigenbaum E A (1981) The handbook of Artificial Intelligence vol 1 Pitman

Barr A and Feigenbaum E A (1982) The handbook of Artificial Intelligence vol 2 Pitman

Bartram D, Beaumont J G, Cornford T, Dann P L and Wilson S L (1986) Recommendations for the design of software for computer based assessment British Psychological Society

Basden A (1983) On the applications of

expert system IJMMS vol 19 p461-477

Bateman D (1987) Pupil use of a knowledge based system IN Nicol (1987)

Becker J (1985a) Expert systems take off at NASA Expert Systems User April 1985 p12

Becker J (1985b) The West 'is ahead' in the appliance of science Expert Systems User April 1985 p22

Becker L A (1987) A framework for intelligent instructional systems: an AI machine learning approach Programmed Learning and Educational Technology vol 24-2 May 1987 p128-136

Beerel A C (1987) Expert Systems: strategic implications and applications Ellis Horwood

Bellman R (1978) An introduction to AI:can computers think ? Boyd and Fraser

Bennett J S and Engelmore R S (1979) SACON: a knowledge-based consultant for structural analysis Proc 6th IJCAI p47-49

Berliner H J (1981) Search versus knowledge IN Elithorn and Banerji (1984)

Bernold T and Albers G (1985) AI: towards practical application North Holland

Berry A (1983) The super-intelligent

machine Jonathan Cape

Berry D and Broadbent D (1986) Expert Systems and the MMI Expert Systems vol 3-4 Oct 86

Beynon M L (1985) An investigation into the use of CAL and the dialogue strategy in a tutorial context JCAL vol 1-1 p15-24

Beynon-Davies P (1988) Expert database systems for the commercial sector Computing Techniques July/Aug 88 p25-27

Bignold P (1986) Expertech Xi in a comprehensive school maths and computer studies IN Thorne (1986)

Blake D V and Uttley A M (1959) Proceedings of the symposium on mechanisation of thought processes HMSO

Blakemore C (1977) Mechanics of the mind Cambridge University Press

Bliss J and Ogborn J (1979) The analysis of qualitative data European Journal of Science Education vol 1-4 p427-440

Bloom B (1956) Taxonomy of educational objectives McKay Co Inc

Bobrow D G and Collins A M (1975) Representation and understanding: studies in

cognitive science Academic Press
Boden M (1977) Artificial intelligence and natural man Harvester Press
Boden M (1983) The educational implications of AI Cognitive Science Research Paper 017 Sussex University
Boden M (1987) Artificial intelligence and natural man (2nd edition) MIT Press
Bolles R (1979) Learning theory Holt, Reinhart and Winston
Boose J H (1985) Methodology for knowledge elicitation, testing, combination and expert system delivery IJMMS vol 23 p495-525
Borman L and Curtis B (1985) Proc CHI '85 Conference on human factors in computing systems ACM
Boseley S (1987) Land of the rising sum Guardian 26/2/87
Brachman R J (1977) What's in a concept: structural foundations of semantic networks IJMMS vol 9 1977 p127-152
Brachman R J (1979) On the epistemological status of semantic networks IN Findler (1979)
Brain K and Brain S (1984) Artificial

Intelligence on the BBC and Electron
Sunshine Books

Bramer M A (1980) A survey and critical review of expert system research Open University

Bramer M A (1984) Research and Development in Expert Systems Cambridge University Press

Bramer M A (1987) Research and Development in Expert Systems III Cambridge University Press

Bramer M A and Bramer D (1984) The fifth generation: an annotated bibliography
Addison Wesley

Bratko I (1986) Prolog programming for AI
Addison Wesley

Bray D W (1984) Using personal computers at the College level IEEE Computer April 1984
p36-43

Bretz R (1971) A taxonomy of communications media Educational Technology Publications

Briggs J H, Dean J and Nicol J (1984)
PLACES and LINX PEG-Exeter University of Exeter

Briggs J H (1984) microProlog rules Logic Programming Associates

Briggs J H (1986) Applications of Expert Systems in Further Education ITDU Kingston College

Briggs J H (1987) Expert Systems in Further Education: A starter pack ITDU Kingston College

Briggs J H (1988) Learning with expert systems FEU

Bronowski J (1973) The ascent of man BCA

Broomfield B (1987) Logic and the rules of law Computer Guardian 26/3/87 p10

Brough D, Clark K L, McCabe F G and Mellish C S (1985) microProlog on the BBC microcomputer Acornsoft

Brough D (1987) The application of IKBS research in education IN Nicol (1987)

Brown J S and Burton R R (1982) An investigation of computer coaching for informal learning activities IN Sleeman and Brown (1982)

Brown J S, Burton R R and de Kleer J (1982) Pedagogical natural language and knowledge engineering techniques in SOPHIE IN Sleeman and Brown (1982)

Brown R (1985) The electronic expert teams up with human skills Computing the Magazine 21/2/85 p10

Bruner J S (1966) Toward a theory of instruction Harvard University Press

Bruner J S (1971) The relevance of education Allen and Unwin

Bryant D (1979) The psychology of resistance to change Management Services vol 23-3 March 79

Bryant N (1988) Managing expert systems John Wiley

Buchanan B (1982) New research on Expert Systems Machine Intelligence vol 10 p269-299

Buchanan B (1986) Expert Systems: working systems and the literature Expert Systems vol 3-1 Jan 86 p32-51

Buchanan B et al (1969) Heuristic Dendral: a program for generating explanatory hypotheses in organic chemistry Machine Intelligence 4

Buchanan B and Feigenbaum E (1978) Dendral and Meta Dendral: their applications dimension AI vol 11 p5-24

Buchanan B and Shortliffe E H (1984) Rule-based expert systems Addison Wesley

Bullinger H J, Protonatorius E N, Bouwhuis D and Reim F (1988) Proceedings of the First European Conference on Information Technology for Organisational Systems Athens, Greece

Bundy A, Burstall R M, Weir S and Young R M (1978) AI: an introductory course Edinburgh University Press

Bundy A (1984) Intelligent front ends IN Bramer (1984)

Bundy A (1987) AI bridges and dreams AI & Society vol 1-1 p62-71

Burton R R (1982) Diagnosing bugs IN Sleeman and Brown (1982)

Burton R R and Brown J S (1979) An investigation of computer coaching for informal learning activities IJMMS vol-11 p5-24

Butcher B T (1973) Human intelligence Harper and Row

Buxton L (1981) Do you panic about maths ? Heinemann

Calderhead J (1984) Teachers' classroom decision making Holt, Rinehart and Winston

Campbell J (1984) Three uncertainties of AI IN Yazdani and Narayanan (1984)

Campbell J and Steels L (1985) Progress in AI Ellis Horwood

Carbonell J (1970) Mixed initiative man-computer instructional dialogues Bolt, Beranek and Newman

Carbonell J (1983a) The XCALIBUR Project Proc 6th IJCAI

Carbonell J (1983b) Learning by analogy: formulating and generating plans from past experience IN Michalski, Carbonell and Mitchell (1983)

Carbonell J (1983c) Deviation analogy in problem solving and knowledge acquisition IN Michalski, Carbonell and Mitchell (1983)

Carlson C (1965) Change processes in public schools University of Oregon Press

Casey C (1986) Simple prolog and simple chemistry Computer Education

CCTA (1984) Expert Systems: a survey of projects in the UK HM Treasury Central Computer and Telecommunications Agency

Chadwick M and Hannah J A (1986) Expert Systems for personal computers Sigma Press
Chalmers, Crawley and Rose (1971) Biological bases of behaviour Open University Press
Chandler D (1984) Young learners and the microcomputer Open University Press
Chandrasekaran B (1983) Expert Systems: matching techniques to tasks IN Reitman (1984)
Charniak E (1978) On the use of frame knowledge AI vol 11-3 p225-266 1978
Charniak E, Riesbeck C and McDermott V D (1983) AI programming Lawrence Erlbaum
Chase L and Simon H A (1974) Cognitive Psychology 4
Chi M T H, Feltovich P J and Glaser R (1981) Categorisation and representation of physics problems by experts and novices Cognitive Science vol 5 p121-152
Chorafas D N (1987) Applying expert systems in business McGraw Hill
Chorover S L (1984) Cautions on computers in education BYTE June 1984 p223-226
Christian-Carter J (1987) The writing of a

knowledge based system for Health Education IN Nicol (1987)

Clancey W J (1982) Tutoring roles for guiding a case method dialogue IN Sleeman and Brown (1982)

Clancey W J (1983a) The advantages of abstract control knowledge in expert system design Proc AAAI 83 p74-78

Clancey W J (1983b) GUIDON Journal of Computer Based Instruction vol 10-1 p8-15

Clancey W J (1984) Methodology for building an intelligent tutoring system IN Kintsch (1984)

Clancey W J (1986) From Guidon to Neomycin and Heracles in twenty short lessons Stanford Knowledge Systems Laboratory working paper 86-11

Clancey W J and Letsinger R (1981) Neomycin: reconfiguring a rule-based expert system for application to teaching Proc 7th IJCAI p829-836

Claridge J and Nicol J (1986) Xi: a critical review IN Thorne (1986)

Clark K L, Ennals R and McCabe F G (1982) A

Micro-Prolog primer Logic Programming Associates

Clark K L and McCabe F G (1984) microProlog: programming in logic Prentice Hall

Clark S (1986) Computer Weekly 2/10/86 p10

Clarke A C (1968) 2001: a space odessey (Hutchinson)

Clarkson D (1986) When IT takes a very long lunch in Abilene Computing 27/3/86 p19

Clement, Kurland, Mawby and Pea (1986) Analogical reasoning and computer programming Educational Computing Research vol 2-4

Clocksin W F and Mellish C S (1981) Programming in PROLOG Springer-Verlag New York

CNAA (1981) Open Learning CNAA publication 1a/33

Cockcroft W H (1982) Maths Counts HMSO

Cohen P R and Feigenbaum E A (1983) The handbook of Artificial Intelligence vol 3 Pitman

Cohen P R and Grinberg M R (1983) A theory of heuristic reasoning about uncertainty AI Magazine Summer 83 p17-24

Colby K M, Weber S and Hilf F D (1971)
Artificial paranoia Artificial Intelligence
2

Colby K M (1975) Artificial paranoia
Pergamon Press

Colley S (1984) People Computing the
Magazine 13/12/84 p28

Collins H M, Green R H and Draper R C (1985)
Where's the expertise ? Expert systems as a
medium of knowledge transfer IN Merry (1985)

Computing (1987) September 17th 87 p20

Conlon T (1985) Learning micro Prolog a
problem-solving approach Addison Wesley

Cooley M (1987) Human centred systems: an
urgent problem AI & Society vol 1-1 p37-46

Coombs M J and Alty J L (1981) Computing
skills and the user interface Academic Press

Cooper W S (1984) Bridging the gap between
AI and IR IN Van Rijsbergen (1984)

Coopers & Lybrand (1985) A challenge to
complacency: a report to the Manpower
Services Commission and the National Economic
Development Office

Copi I (1982) Introduction to logic Macmillan

Cotterell A, Ennals R and Briggs J (1988) Advanced information technology in education and training Edward Arnold

Cowan J (1986) Learner-centred learning: the key issues IN Percival et al (1987)

Coyne R D, Rosenman M A, Radford A D, Balachandran B M and Gero J S (1988) Knowledge-based design systems Addison Wesley

Crowder N A (1959) Automatic tutoring by means of intrinsic programming IN Galanter (1959)

D'Agapeyeff A (1983) Expert systems, fifth generation and UK suppliers NCC publications

D'Agapeyeff A (1984a) A report to the Alvey Directorate on a short survey of expert system in UK business Supplement to Alvey News no.4 April 1984

D'Agapeyeff A (1984b) Expert System interview Expert Systems vol 1-2 p129-135

D'Agapeyeff A (1987) A report to the Alvey

Directorate on the second short survey of Expert Systems in UK business Consultants in IT London

Daines D R (1984) Databases in the classroom Castle House Publications

Davies P M (1969) The hydraulic theory of education IN O'Shea and Self (1983)

Davies S (1986) Xi in the comprehensive school curriculum IN Thorne (1986)

Davis R (1982) Expert Systems: where are we and where do we go from here AI Magazine Spring 1982

Davis R (1984) Amplifying expertise with expert systems IN Winston and Prendergast (1984)

Davis R and Buchanan B (1977) Meta-level knowledge: overview and application Proc IJCAI 1977 p920-928

Davis R and King J (1976) An overview of production systems Machine Intelligence 8 John Wiley

Davis R and Lenat D B (1982) Knowledge-based systems in AI McGraw Hill

Dawkins P (1986) Expert Systems 85; The business tutorial First steps:a strategy for success Expert Systems SIG newsletter 15/5/86 p18-20

deBono E (1976) Teaching thinking Temple Smith

deBono E (1982) deBono's thinking course BBC

Dede C (1986) A review and synthesis of recent research in ICAI IJMMS vol 24-4 p329-353

Deeson E (1984) 1994-year of great big brother Computers in Schools vol 6-4 Heinemann

deJong G (1983) Acquiring schemata through understanding and generalising plans Proc. IJCAI 1983 p462-464

Dennett D C (1979) Brainstorms: philosphical essays on mind and psychology Harvester Press

Descartes R (1970) Discourse on method Penguin

Diaper D (1986) Identifying the knowledge requirements of an expert system's natural language processing interface IN Harrison and Monk (1986)

Dixon M (1986) The lessons training offers to expert systems Expert Systems User Sept 1986 p18-19

Dixon M (1988) The use of knowledge-based systems in training IN Bullinger et al (1988)

Donaldson M (1963) A study of childrens thinking Tavistock Publications

Dreyfus H L (1979) What computers can't do Harper and Row

Dreyfus H L and Dreyfus S E (1986) Mind over machine Basil Blackwell

Dreyfus H L and Dreyfus S E (1987) The mistaken psychological assumptions underlying belief in expert systems IN Costall and Still (1987)

Duda R and Gaschnig J et al (1978) Model design in the PROSPECTOR consultant system for mineral exploration IN Michie (1980)

Duke J (1985) Information management and the consultant: towards the infallible expert ASLIB Proceedings vol 37-3 March 1985

Dungan C W and Chandler J S (1985) Auditor: a microcomputer based expert system to support auditors in the field Expert Systems vol 2-4 Oct 85 p210

Dunn S and Morgan V (1987) The impact of the computer on education: a course for teachers Prentice Hall International

Durant D (1987) 5G IT and Learning vol 10-1 p17-20

Durham T (1985) Finding new reasons to believe in expert systems Computing 14/11/85 p2-4

Durham T (1986) Taking a look inside the untidy mind Computing 23/10/86 p26-27

Durham T (1987) Neural brainwaves break new ground Computing 9/4/87 p28-29

Dyer R (1988) Can we talk ? AI applications in natural resource management vol 2-4 p57-59

Eary J (1987) Training by stealth Interactive Learning International vol 4-3&4 p90-94

Eisenstadt M and Brayshaw M (1987) Integrated Prolog environment IN Nicol (1987)

Elithorn A, Cooper R and Telford A (1981) Benchmark and yardstick problems: a systematic approach IN Elithorn and Banerji

Elithorn A and Banerji R (1984) Artificial and human intelligence Elsevier

Ellingham D (1982) Managing the microcomputer in the classroom CET

Ellingham D (1984) Handbook of Primary Education and computing Castle House

Ellis A (1974) The use and misuse of computers in education McGraw Hill

Else L (1985) Oil majors show natural reserve in DP exploration Computing The Magazine 14/3/85 p12

Engelmore R and Morgan T (1988) Blackboard systems Addison Wesley

Ennals R (1983) Beginning micro Prolog Ellis Horwood

Ennals R (1984) microProlog in the classroom 1980-1984 Computers in Schools vol 7-4 p116-118

Ennals R (1986a) A way forward for advanced IT: SHI a strategic health initiative IN Gill (1986)

Ennals R (1986b) New infrastructure for research and technology transfer Future Computing Systems vol 1-1 p13-29

Ennals R (1987) Issues of curriculum in the F.E. sector IN Nicol (1987)

Ennals R and Cotterell A (1985) Fifth generation computers, their implications for F.E. DES Further Education Unit

Ennis R L et al (1986) A continuous real-time Expert System for computer operations IBM Journal of R&D vol 30-1 Jan 86 p14-27

Ericsson K A and Simon H A (1984) Protocol analysis: verbal reports as data MIT Press

Ernst C (1988) Management expert systems Addison Wesley

Essinger J (1988) Computers in financial trading Elsevier Advanced Technology Publications

Estes W K (1978) Handbook of learning and cognitive processes Lawrence Erlbaum Associates

Estes W K (1982) Learning, memory and intelligence IN Sternberg (1982)

Evans C (1979) The mighty micro Gollancz

Evans C (1981) The making of the micro Gollancz

Evans N (1986) The future of the microcomputer in schools Macmillan Education

Expert Systems (1985) Vol 2-4 Oct 85 p190

Expert Systems (1986) Vol 3-2 April 86 p120

Feigenbaum E A (1979) Themes and case studies of knowledge engineering IN Michie (1979)

Feigenbaum E A (1980) Knowledge engineering: the applied side of AI Stanford University Dept of Computer Science Report STAN-CS-80-812

Feigenbaum E A (1982) Knowledge engineering for the 1980's Stanford University

Feigenbaum E A and Feldman J (1963) Computers and thought McGraw Hill

Feigenbaum E A and McCorduck P (1984) The Fifth Generation Michael Joseph

Feinstein J L and Siems F (1985) EDAAS Expert Systems vol 2-2 April 85 p72

Findler N V and Meltzer B (1971) AI and heuristic programming Edinburgh University Press

Findler N V (1979) Associative Networks: Representation and use of knowledge by computers Academic Press

Fischler M A and Firschein O (1987) Intelligence: the eye, the brain and the computer Addison Wesley

Fisher G, Lemke A and Scwab T (1985) Knowledge based systems IN Borman and Curtis (1985)

Fitter M and Green T R G (1979) When do diagrams make good computer languages ? IJMMS vol 11 p235-261

Ford L (1986) Instruction and support for computer applications ISCA/IR/1 University of Exeter

Ford N (1987) How machines think John Wiley

Fordyce D (1986) Thinking: can it improve the quality of student learning ? IN Percival et al (1987)

Foremski T (1986) AI firms hope for jam tomorrow Computing 6/11/86 p20

Forsyth R (1984a) Expert Systems: principles and case studies Chapman and Hall

Forsyth R (1984b) The silicon valley of the dolls IN Yazdani and Narayanan (1984)

Forsyth R (1988) Software review: Leonardo Expert Systems vol 5-2

Forsyth R and Naylor C (1985) The

Hitch-Hikers Guide to AI Chapman and Hall/Methuen

Forsyth R and Rada R (1986) Machine learning: applications in Expert Systems and information retrieval Ellis Horwood

Fox J (1980) Making decisions under the influence of memory Psychological Review vol 87-2 p190-211

Fox J (1983) Expert Systems '83 Cambridge University Press

Fox J (ed) (1984) Infotech State of the Art Report on Expert Systems Infotech Pergamon

Fox M S (1986) Industrial applications of AI Robotics 2 p301-311 Elsevier Science Publishers

Fox M S and Smith S F (1984) ISIS - a knowledge-based system for factory scheduling Expert Systems vol 1-1 p25

French P D (1987) Intelligent training IN Nicol (1987)

Freyenfield W A (1984) Decision support systems NCC Publications

Friedland P (1981) Acquisition of procedural knowledge from domain experts p856-861 Proc 7th IJCAI Vancouver British Columbia 1981

Frost and Sullivan (1985) Expert Systems in Europe Frost and Sullivan London

Fuchi K (1983) The direction the FGCS project will take New generation computing vol 1 p3-9

Gagne R M (1965) The conditions of learning Holt

Galanter E (1959) Automatic teaching: the state of the art Wiley

Galanter E (1983) Kids and computers Kingfisher

Gammack J G and Young R M (1984) Psychological techniques for eliciting expert knowledge IN Bramer (1984)

Ganascia J and Kodratoff Y (1985) Symbolic uncertain inference: a study of possible modalities IN Merry (1985) p169

Gardner A (1987) An AI approach to legal reasoning MIT Press

Garland R (1982) Microcomputers and children in the Primary school Falmer Press

Geyer T (1988) Learning through computer simulation - a possible contribution from AI Computer Education Feb 88 p11-12

Gill K S (1986) Artificial Intelligence for Society John Wiley

Gill P (1983) Microcomputer learning in the Primary school Ward Lock Ed

Glassup B (1985) Infomatics November 85 p71

Goble T (1987) The use of Prolog in geography teaching IN Nicol (1987)

Goldberg D, Fund Z and Scherz Z (1987) Mistakes and misconceptions in learning Prolog IN Nicol (1987)

Goldstein I (1979) The genetic graph: a representation for the evolution of procedural knowledge IJMMS vol 11-1 p51-78

Goodall A (1985) The guide to Expert Systems Learned Information

Goodyear P (1984) LOGO: a guide to learning through programming Ellis Horwood

Goodyear P (1987) Expert systems and intelligent tutoring: some issues in the engineering of pedagogic knowledge CERCLE TR 35 University of Lancaster

Gottinger H W (1984) Hazard: an Expert Systems for screening environmental chemicals on carcinogenity Expert Systems vol 1-2 p169

Gotts N M, Hunter J R W and Sinnhuber R K E W (1984) An intelligent model-based system for diagnosis in cardiology: a research proposal University of Sussex AIMG-7

Gronlund N (1970) Stating behavioural objectives for classroom instruction Macmillan

Guest D (1987) PC Magazine May 87

Guilford J P, Kettner N M and Christensen P E (1956) The nature of the general reasoning factor Psychological Review 63 p169-172

Guilfoyle C and Jeffcoate J (1987) Expert Systems in banking and securities Ovum

Guy K (1987) The use and development of Expert Systems in the UK Alvey News April 84 p20-21

Hafner C D (1981) An information retrieval system based on a computer model of legal knowledge UMI Research Press

Hall R (1986) Learning by failing to explain Proc. AAAI-86 p568-573

Hammond D (1986) HP West country source for real systems Expert Systems User Jan 1986 p12

Hammond K J (1987) Learning and reusing explanations IN Langley (1987)

Hammond R (1983) Computers and your child Century

Harmon P and King D (1985) Expert Systems AI in business John Wiley

Harrison M D and Monk A F (1986) People and computers: designing for usability Cambridge University Press

Hart A (1986) Knowledge acquisition for Expert Systems Kogan Page

Hart P E (1980) What's preventing the widespread use of expert systems ? p11-14 Position paper Expert Systems Workshop San Diego, California

Hartley R (1973) The design and evaluation of an adaptive teaching system IJMMS 5-2

Hartley R (1981) How expert should an expert be ? p862-867 Proc 7th IJCAI Vancouver British Columbia

Hartley R (1982) Micros in the maths classrooms CET

Harvey B (1984) Why LOGO ? IN Yazdani (1984)

Hassell D (1987) Using ADEX advisor as a tool for qualitative modelling: some experiences from secondary school trials IN Nicol (1987)

Hawkins D (1983) An analysis of expert thinking IJMMS vol 18 p1-47

Hawkridge D (1983) New information technology in education Croom Helm

Hayes J and Michie D (1983) Intelligent systems: the unprecedented opportunity Ellis Horwood

Hayes J, Michie D and Mikilich L I (1979) Machine Intelligence 9 Ellis Horwood

Hayes P J (1977) On semantic nets, frames and associations Proc 5th IJCAI 1977

Hayes-Roth F (1983) An overview of Expert Systems IN Hayes-Roth et al (1983)

Hayes-Roth F, Waterman D A and Lenat D (1983) Building expert systems Addison Wesley

Hayward S A (1984) Is a decision tree an Expert System ? IN Bramer (1984)

Heaford J M (1983) The myth of the learning machine Sigma Technical Press

Hewett J and Sasson R (1986) Expert Systems 1986; USA and Canada Ovum

Hewett J, Timms S and d'Aumale G (1986) Commercial Expert Systems in Europe Ovum Ltd

Hilgard E R and Bowe G H (1966) Theories of learning Appleton Century Crofts

Hirst and Peters (1970) The logic of education Routledge and Kegan Paul

Hockaday M (1986) AI Pack beats city tipsters Datalink 14/4/86

Hofstadter D (1977) Godel, Escher, Bach: an eternal golden braid Harvester Press

Hogger C (1984) Introduction to logic programming Academic Press

Holtzmann W H (1970) Computer assisted instruction, testing and guidance Harper and Row

Hooper R and Toye I (1980) CAL in the UK CET

Howe J (1978) AI and CAL: ten years on IN Rushby (1981)

Howe J (1987) IKBS - setting the scene IN O'Shea et al (1987)

Howe J and Ross P (1984) Microcomputers in secondary education Kogan Page

Hudson K (1984) Introducing CAL Chapman and Hall Computing

Hughes S (1986) Question classification in rule-base systems IN Bramer (1987)

Hughes and Hughes (1965) Learning and teaching Longman

Humphries C (1986) Training Access Point: a feasibility study carried out by the Council for Educational Technology for the UK Manpower Services Commission

Hunt E B (1975) Artificial intelligence Academic Press

Hutchins W J (1986) Machine translation: Past, present and future Ellis Horwood

Illich I (1971) Deschooling society Harper and Row

Inhelder B and Piaget J (1958) The growth of logical thinking from childhood to adolescence Basic Books

Innocent P (1982) Towards self adaptive interface systems IJMMS vol 16

Intelligence (1921) Intelligence and its measurement: a symposium Journal of Educational Psychology 12

Ishizuka M (1984) Japanese work in Expert Systems Expert Sys vol 1-1 p51-56

Jacob V S, Gaultney L D and Salvendy G (1986) Strategies and biases in human decision making and their implications for expert systems Behaviour and Information Technology vol 5-2 p119-140

James M (1984) AI in BASIC Newnes Technical

Johnson P (1983) What kind of expert should a system be ? The Journal of Medicine and Philosophy vol 8 1983 p77-97

Johnson T (1984) The commercial applications of Expert Systems technology Ovum Ltd

Johnson T (1986) Natural language computing: the commercial applications Ovum

Johnson-Laird P N and Wason P C (1977) Thinking: readings in cognitive science Cambridge University Press

Johnston R (1985a) Shells are not enough for Plessey Expert Systems User Sept 1985 p12

Johnston R (1985b) How an expert system was built without an expert Expert Systems User Nov 85 p16-17

Johnston R (1985c) How Wall St is banking on the Expert Systems Computer Guardian 21/11/85 p13

Jones R (1981) Microcomputers: their uses in primary Schools CET

Jones R (1986a) Commercial expert systems: now to avoid the pitfalls Data Processing vol 28-3 April 86 p115-119

Jones R (1986b) Breaking the spell of Expert Systems sorcery Computer Weekly 5/6/86 p34

Jones T (1980) Microelectronics and society Open University Press

Jones V and Davies K (1986) A taxonomy of application areas for expert systems in business University of Stirling Dept of Computer Science Report TR31

Kelly G A (1955) The psychology of personal constructs Norton

Kemp R, Stewart T and Boorman A (1988) Improving the expert system interface AI applications in natural resource management vol 2-4 p48-53

Kempf K G (1984) Manufacturing and AI IN

Bernold and Albers (1985) p1-17

Kidd A and Cooper M B (1983) Man-machine interface for an expert system IN Fox (1983)

Kidd A and Welbank M (1983) Knowledge Acquisition IN Fox (1984)

Kidd A (1985) Human factors in expert systems Data Processing vol 27-4 p15-17

Kiesler S B and Sproull L S (1987) Computing and change on campus Cambridge University Press

Kimball R A (1982) A self-adapting, self improving tutor for symbolic integration IN Sleeman and Brown (1982)

Kintsch W (1984) Methods and tactics in cognitive science Lawrence Erlbaum

Klahr D (1976) Cognition and instruction Lawrence Erlbaum Associates

Klahr P and Waterman D (1986) Expert Systems techniques, tools and applications Addison Wesley

Kleinmuntz B (1968) Formal representation of human judgement Wiley p148-186

Knasel T M (1986) AI in manufacturing: forecasts for the use of AI in the USA Robotics vol 2-4 Dec 86 p357-362 Elsevier

Science Publishers

Kodratoff Y (1988) Introduction to machine learning Pitman

Kolb D A (1984) Experiental learning: experience as the source of learning and development Prentice Hall

Kolodner J L (1983) Towards an understanding of the role of experience in the evolution from novice to expert IJMMS 19 p497-518

Kowalski R (1979) Logic for problem solving North Holland

Kowalski R (1984) Logic as a computer language for children IN Yazdani (1984)

Kowalski R (1987) Logic programming in AI and software engineering IN O'Shea et al (1987)

Kowalski R and Sergot M (1985) A logic-based calculus of events Imperial College London Dept of Computing Report

Krutch J (1981) Experiments in AI for small computers Sams and Co

Langley P (1982) Strategy acquisition governed by experimentation IN Campbell and Steels (1985)

Langley P, Bradshaw G L and Simon H A (1982) Studying chemistry with the BACON system IN Michalski et al (1982)

Langley P (1987) Proceedings of the 4th machine learning workshop Morgan Kaufmann

Lantz B S, Bregar W S and Farley A M (1983) An intelligent CAI system for teaching equation solving Journal of Computer Based Instruction Summer 1983 vol 10 p35-42

Large P (1984) The microrevolution revisited Frances Pinter

Lavic N et al (1985) Kardio-E: an Expert Systems for electrocardiographic arrhythmias Expert Systems vol 2-1 p46

Lawler R (1984) Designing computer-based microworlds IN Yazdani (1984)

Lawler R (1985) Computer experience and cognitive development Ellis Horwood

Lehman M (1988) The perils of polluted software Computing 2/11/88 p26

Leinhardt G and Greeno J (1986) The cognitive skill of teaching Journal of

Educational Psychology vol 78 p78-95

Leith P (1986) Fundamental errors in legal logic programming The Computer Journal vol 29-6 p545-552

Lenat D B (1977a) Automated theory formation in maths Proc 5th IJCAI p 833-842

Lenat D B (1977b) The ubiquity of discovery Artificial Intelligence vol 9-3 p257-285

Lenat D B (1981) The nature of heuristics Artificial Intelligence vol 19

Lenat D B (1983a) The role of heuristics in learning by discovery: 3 case studies IN Michalski, Carbonell and Mitchell (1983)

Lenat D B (1983b) EURISKO: a program that learns new heuristics and domain concepts Artificial Intelligence vol 21

Lenat D B et al (1983c) Cognitive economy in a fluid task environment IN Michalski (1983)

Lesgold A M (1978) Cognitive psychology and instruction Plenum

Levesque H J (1984) Foundations of a functional approach to knowledge representation AI (23) 1984

Lewis R and Tagg E D (1980) Computer

assisted learning North Holland

Liebeck P (1984) How children learn mathematics Penguin

Lieberman H (1986) An example based environment for beginning programmers Instructional Science vol 14 p277-292

Liebowitz J (1987) Common fallacies about expert systems Computers and society vol 16-4 p28-33

Lighthill J (1973) Artificial Intelligence: a paper symposium Science Research Council HMSO

Lindsay R, Buchanan B G, Feigenbaum E A and Lederburg J (1980) Application of AI for organic chemistry: the DENDRAL project McGraw Hill

Loftus C R and Loftus E F (1976) Human memory: the processing of information Halstead

Longuet-Higgins C (1987) Mental processes: studies in cognitive science MIT Press

Looney C G and Alfize A R (1986) Towards Expert System on a chip SIGART newsletter 98 October 1986 p26-28

Lucash R M (1986) Legal liability for

malfunction and misuse of Expert Systems SIGCHI Bulletin vol 18-1 July 86 p35-43

Maddison A (1982) Microcomputers in the classroom Hodder and Stoughton

Mathieson R (1986) Tomorrows accountant - expert aided ? The Accountant's Magazine April 86 p22

Mason E J et al (1983) Three approaches to teaching and learning in education: behavioural, Piagetian and Information processing Instructional Science vol 12 p219-241

Matz M (1982) Towards a process model for High school errors IN Sleeman and Brown (1982)

Maugh T H (1986) Associative memory technology: closer to function of brain IEEE Expert Summer 1986 p99

McCarthy J (1958) Programs with common sense IN Blake and Uttley (1958)

McCarthy S (1986) Xi in the Primary curriculum IN Thorne (1986)

McHale J (1976) The changing information environment Paul Elek London

McLaren R (1984) Expert Ease user manual Intelligent Terminals

McCorduck P (1979) Machines who think Freeman and Co

McCracken D and Akscyn R (1984) Experience with ZOG human computer interface system IJMMS vol 21 p293-310

McDermott J (1984) R1 revisited: four years in the trenches AI Magazine vol 5-3 Fall 84 p21-24

McNally D W (1974) Piaget, education and teaching New Educational Press

McSharry D and Fullerton K (1985) Preceptor: a shell for medical expert systems and its application in a study of prognostic indices in stroke Expert System vol 2-3 p140

Meighan R (1981) A sociology of education Holt

Merrill M D (1983) Component display theory IN Reigeluth (1983)

Merry M (1985) Expert Systems '85 the proceedings of the fifth technical conference of the BCS specialist interest group on Expert Systems Cambridge Univ Press

Michaelsen R H, Michie D and Boulanger A (1985) The technology of expert systems Byte April 1985 p303-312

Michalski R S (1983) Proceedings of the International Machine Learning workshop University of Illinois

Michalski R S and Chilausky R L (1980) Knowledge acquisition by encoding expert rules versus computer induction from examples: a case study involving soybean pathology IJMMS 1980 vol 12 p63-87

Michalski R S and Stepp R E (1983) Learning from observations: conceptual clustering IN Michalski, Carbonell and Mitchell (1983)

Michalski R S, Carbonell J G and Mitchell T M (1983) Machine learning: the AI approach Tioga Publishing Co

Michie D (1974) On machine intelligence Edinburgh Univ Press

Michie D (1979) Expert systems in the micro-electronic age Edinburgh Univ Press

Michie D (1980) Social aspects of AI IN Jones (1980)

Michie D (1982a) Introductory readings in Expert systems Gordon and Breach

Michie D (1982b) Machine intelligence and related topics Gordon and Breach

Michie D (1986) On machine intelligence 2nd edition Ellis Horwood

Michie D and Johnston R (1985) The creative computer Pelican

Mill J (1985) Norway strikes oil with offshore AI Computing the Magazine 14/3/85 p12

Mill J (1986) Hatching a future for expert system shells Computing 13/2/86 p3-5

Miller G A (1956) The magical number seven, plus or minus two Psychological Review 63 p81-97

Miller G A, Galanter E and Pribram K H (1971) Plans and the structure of behaviour Holt

Miller P L (1984) ATTENDING: a critiquing approach to expert computer advice Pitman

Minsky M (1968) Semantic information processing MIT Press

Minsky M (1975) A framework for representing knowledge IN Winston (1975)

Minsky M and Papert S (1969) Perceptrons:

an introduction to computational geometry
MIT Press

Mitchell T M (1982) Learning by experimentation: acquiring and refining problem solving heuristics IN Michalski et al (1983)

Mitchell T M, Mahadevan and Sternberg L I (1985) LEAP: a learning apprentice for VLSI design Proc. IJCAI 1985 p573-580

Mitchell T M, Carbonell J G and Michalski R S (1986) Machine learning: a guide to current research Kluwer Academic Publishers

Monod J (1972) Chance and necessity Collins

Mooney R J (1987) EBL: a general learning mechanism and its application to several complex domains Workshop on complex learning Lancaster University April 1987

Moralee S (1988) Research and development in expert systems IV Cambridge University Press

Moses J (1971) Algebraic simplification: a guide for the perplexed Communications of Association for Computing Machinery 14

Mostow D J (1983a) Machine transformation of advice into a heuristic search procedure IN Michalski, Carbonell and Mitchell (1983)

Mostow D J (1983b) International learning workshop: an informal report SIGART newsletter 86 p367-403

Moto-Oka T (1982) Fifth generation computing systems: proceedings of the International Conference on Fifth Generation Computer Systems North-Holland

Munro (1969) Psychology and education of the young Heinemann

Murray D and Bevan N (1984) The social psychology of computer conversations Interact '84 First IFIP Conference on Human Computer Interaction

Myers E (1986) Not for everyone Datamation vol 32-10 15/5/86 p28-32

Naughton J (1986) AI: applications to training Open University

Naylor C (1983) Build your own expert system Sigma Technical Press

NCC (1985) Expert Systems Starter Pack NCC Publications

NCC (1987) Expert Systems Resource Pack
NCC Publications

Negrotti N and Bertasio B (1986) The Archimedes syndrome: cultural premises and AI technology IN Gill (1986)

Newell A (1968) Heuristic programming: Ill-structured problems IN Aronofsky (1969)

Newell A (1980) Duncker on thinking: an enquiry into progress CMU Report CS-80-151

Newell A (1984) Speech: the natural modality for man-machine interaction ? Interact '84 First IFIP Conference on Human Computer Interaction

Newell A and Simon H A (1963) GPS: a program that simulates human thought IN Feigenbaum and Feldman (1963)

Newell A and Simon H A (1972) Human problem solving Prentice Hall

Newton W (1988) Computer Assisted Guidance Project CET News 31 Spring 88

Nicol J and Dean J (1984) Pupils, computers and history teaching IN Yazdani (1984)

Nicol J, Dean J and Briggs J H (1986) Fifth generation computing in the classroom Computers in Schools vol 8-3 p75-80

Nicol J (1987) Proceedings of the Prolog Education Group 2nd International Conference Exeter University

Nilsson N (1971) Problem solving methods in AI McGraw Hill

Nilsson N (1982) Principles of AI Springer Verlag

Norman A and Cattell G (1983) LISP on the BBC microcomputer Acornsoft

Norman D A (1978) Notes towards a complex theory of learning IN Lesgold (1978)

Norman D A (1981) A psychologist views human processing: human errors and other phenomena suggest processing mechanisms Proc 7th IJCAI p1097-1101

Norman D A (1984) Stages and levels of human machine dialogue IJMMS vol 21

Norman D A (1985) Twelve issues for cognitive science IN Aitkenhead and Slack (1985)

O'Connell K (1987) Prolog tools for older Primary pupils IN Nicol (1987)

O'Connor D E (1983) Using expert systems to manage change and complexity in manufacturing IN Reitmann (1984)

Ohlsson S (1986) Some principles of intelligent tutoring Instructional Science 14 p293-326

Oliff M (1988) Intelligent manufacturing Addison Wesley

O'Shea T (1981) Intelligent systems in education Infotech state of the Art Report series 9 no. 3 Pergamon Infotech

O'Shea (1982) A self improving quadratic tutor IN Sleeman and Brown (1982)

O'Shea (1987) Machine learning IN O'Shea et al (1987)

O'Shea T and Eisenstadt M (1984) Artificial Intelligence - tools, techniques and applications Harper and Row

O'Shea T and Self J (1983) Learning and teaching with computers Harvester Press

O'Shea T, Self J and Thomas G (1987) IKBS: an introduction Harper and Row

PACTEL (1985) Expert systems survey. A summary of findings PA Computers and Telecommunications London

PACTEL (1987) Expert systems survey PA Computers and Telecommunications London

Paice C (1986) Expert systems for information retrieval ? ASLIB Proceedings vol 38-10 October 86 p343-353

Paine N E (1986) The significance of flexible learning systems IN Percival et al (1987)

Papert S (1980) Mindstorms, children, computers and powerful ideas Harvester Press

Partridge D (1988) To add AI or not to add AI ? Keynote lecture from Expert Systems '88

Pask G and Curran S (1982) Microman Century

Pask G and Scott B C E (1972) Learning stategies and individual competence IJMMS 4

Pateman T (1981) Communicating with computer programs Language and Communication vol 1 p3-12

Pearson A W (1984) Speculations on the future of knowledge engineering in Europe 2 IN Bernold and Albers (1984) p185-187

Percival F, Craig D and Buglass D (1987) Aspects of educational technology Volume XX Flexible learning systems Kogan Page

Piaget J (1971) Science of education and the psychology of the child Longman

Piddock P (1987) Expert systems and education Computer Education February 1987 p8-9

Platt C (1984) Micromania Gollancz

Polya G (1945) How to solve it Princeton Univ Press

Popper K (1974) Objective knowledge Clarendon Press

Quillian M R (1968) Semantic memory IN Minsky 1968

Quinlan J R (1979a) Discovering rules by induction from large collections of examples IN Michie (1979) p168-201

Quinlan J R (1979b) Induction over large databases Stanford Heuristic Programming Project STAN-CS-79-739

Raffan J (1987) KBS and dyslexic pupils IN Nicol (1987)

Raphael B (1976) The thinking computer: mind inside matter W H Freeman

Rasmussen J (1987) When pupils learn

Prolog: difficulties with syntax and semantics IN Nicol (1987)

Rauch-Hindin W (1985) AI in business, science and industry Vol 2 Applications Prentice Hall

Rauch-Hindin W (1986) AI in business, science and industry Vol 1 Fundamentals Prentice Hall

Rees B (1984) AI in a large scale enterprise - the experience of Digital Equipment Corporation IN Bernold and Albers (1985) p67-76

Reichgelt H and van Harmelen F (1985) Relevant criteria for choosing an inference engine in expert systems IN Merry (1985)

Reigeluth C M (1983) Instructional design theories and models: an overview of their current status Erlbaum

Reitmann W (1984) AI applications for business Ablex

Richards M and Underwood K (1984) How should people and computers speak to each other ? Interact '84 First IFIP Conference on Human Computer Interaction

Rissland E (1984) Ingredients of

intelligent user interfaces IJMMS vol 21

Rogers J B (1984) Computer use in Pre-College Education IEEE Computer April 1984 p46-52

Rogers W (1979) Computer aided medical diagnosis: literature review International Journal of Biomedical Computing vol 10 p267-289

Rosenblatt F (1958) The Perceptron, a probabilistic model for information storage and organisation in the brain Psychological Review 65

Ross P (1983) LOGO programming Addison Wesley

Rumelhart D and Norman D (1983) Representation in memory Center for Human Information Processing, California

Rushby N J (1980) An introduction to educational computing Croom Helm

Rushby N J (1981) Selected readings in computer-based learning Kogan Page

Ruthven K (1985) The AI dimension ? IN Smith (1985)

Rutkowska J C (1986) Developmental psychology's contribution to cognitive science IN Gill (1986)

Rychener M D (1985) Expert systems for engineering design Expert Systems vol 2-1 January 1985 p30-44

Sacerdoti E D (1982) Practical machine intelligence Machine Intelligence 10 p241-247

Sagalowicz D (1984) Expert systems in service sectors: use of expert systems in six sample cases IN Bernold and Albers (1985) p77

Sage M and Smith D (1983) Microcomputers in Education: a framework for research SSRC

Sathi A, Fox M S, Greenberg M (1985) Representation of activity knowledge for project management IEEE Transactions on Pattern Analysis and Machine Intelligence vol PAMI-7 no. 5 p531-552

Scanlon E and O'Shea T (1987) Educational Computing John Wiley

Scarr S (1986) Intelligence revisited IN Sternberg and Detterman (1986)

Schank R C (1975) The structure of episodes in memory IN Bobrow and Collins (1975)

Schank R C (1982) Dynamic memory: a theory of reminding and learning in computers and people Cambridge University Press

Schank R C and Colby K M (1973) Computer models of thought and language Freeman

Schank R C et al (1975) Scripts, plans and knowledge Proc 4th IJCAI 1975

Schank R C and Hunter L (1985) The quest to understand thinking Byte April 1985 p143-155

Schools Council (1972) With objectives in mind Macdonald

Schools Council (1980) Learning through science Macdonald

Schubert L K (1976) Extending the power of semantic networks AI 7(2) 1976 p163-198

Schwartz W B (1970) Medicine and the computer: the promise and problems of change New England Journal of Medicine vol 283 p1257-1264

Searle J (1984) Minds, brains and machines The 1984 Reith lectures BBC

Self J (1974) Student models in computer aided instruction IJMMS vol 6 p261-276

Self J (1979) Student models and AI
Computers and Education vol 3-4

Self J (1985a) A perspective on intelligent
CAL Journal of CAL vol 1-3

Self J (1985b) Microcomputers in education:
a critical appraisal of educational software
Harvester Press

Self J (1987a) IKBS in education
Educational Review vol 39-2 p147-154

Self J (1987b) User modelling in open
learning systems CERCLE TR 34 University of
Lancaster

Self J (1988) AI and human learning
Chapman and Hall

Sell C (1984) Expert systems - a practical
introduction Methuen

SERC-DoI (1983) IKBS a programme of action
in the UK vol 2 SERC-DoI

Shallis M (1984) The silicon idol Oxford
Univ Press

Shapiro E (1982) Algorithmic program
debugging MIT Press Cambridge MA

Sharples M (1981) Microcomputers and
creative writing IN Howe and Ross (1984)

Sharples M and Finlayson H (1985) LOGO Software MEP LOGO Pack

Shneiderman B (1980) Software psychology: human factors in computer and information systems Boston

Shorter D (1987) Achievements within the IKBS programme, March 1987 Alvey News April 87 p10-13

Shortliffe E H (1976) Computer based medical consultations: MYCIN American Elsevier

Sibbett R (1987) A Prolog tool for handling structured knowledge IN Nicol (1987)

Simmons M K (1984) AI for engineering design CAE Journal April 84 p75-83

Simon H A (1972) The theory of problem solving Information Processing North Holland

Simons G (1983a) Expert systems and micros NCC Publications

Simons G (1983b) Are computers alive ? evolution and life forms Harvester Press

Skinner B F (1958) Teaching machines Science vol 128

Skinner B F (1971) Beyond freedom and dignity Knopf

Sleeman D and Brown J (1982) Intelligent tutoring systems Academic Press

Sleeman D and Ward R D (1988) Intelligent tutoring systems in training and education: prospects and problems Invited paper at Expert Systems '88

Sloman A (1987) AI languages IN O'Shea et al (1987)

Smallwood R D (1970) Optimal policy regions for computer directed teaching systems IN Holtzmann (1970)

Smart G and Langeland-Knudsen J (1986) The CRI Directory of Expert Systems Learned Information Ltd Oxford

Smith D J (1985) IT and Education: signposts and research directions ESRC

Smith H T and Green T R G (1980) Human interaction with computers Academic Press

Smith R G (1984) On the development of commercial ES AI Magazine vol 5-3 Fall 84 p61-73

Smith S B and Lane R S (1984) An intelligent operator aid for dynamic route planning Expert Systems vol 1-2 p143

Snelbecker G E (1974) Learning theory, instructional theory and psychoeducational design McGraw Hill

Solomon C (1986) Computer environments for children - a reflection on theories of learning and education MIT Press

Soloway E, Rubin E, Woolf B, Bonar J and Lewis Johnson W (1983) MENO-II: an AI-based programming tutor Journal of Computer Based Instruction Vol 10 p20-34

Sparck Jones K (1984) Natural language interfaces for expert systems IN Bramer (1984)

Sparck Jones K and Wilks Y (1985) Automatic natural language parsing Ellis Horwood

Speller G J and Brandon J A (1986) Ethical dilemmas of expert systems Behaviour and Information Technology vol 5-2 p141-143

Sridharan N S (1978) Guest editorial AI vol 11 p1-4

Sriram D and Rychener M D (1986) Expert systems for engineering applications IEEE Software March 86 p3

Stefik M (1978) Inferring DNA structures from segmentation data AI vol 11 p85-114

Stefik M et al (1982) The organisation of expert systems: a tutorial AI vol 18 p135-172

Stefik M and deKleer J (1983) Prospects for expert systems in CAD Computer Design 21/4/83 p65-76

Stern N (1987) The ideas behind educational knowledge engineering IN Nicol (1987)

Sternberg R J (1982) Handbook of human intelligence Cambridge University Press p170-224

Sternberg R J (1986) A framework for understanding conceptions of intelligence IN Sternberg and Detterman (1986)

Sternberg R J and Detterman D K (1986) What is intelligence ? Ablex

Stevens A L and Collins A (1978) The goal structure of a Socratic tutor Bolt, Beranek and Newman

Stevens A (1984) How shall we judge an expert system ? IN Forsyth (1984a)

Stewart A (1986) Japan has a clear vision of the future Expert System User Sept 85 p14

Stewart S D (1986) Expert systems invades military Simulation February 1986 p69

Stock M (1988) Planning expert system projects AI applications in natural resource management vol 2-4 p9-16

Stone J (1988) Case study of the use of expert systems in Further Education physics IN Briggs (1988)

Stonier T (1983) The wealth of information Methuen

Stonier T (1984) The knowledge industry IN Forsyth (1984a)

Stow R, Lunn S and Slatter P (1986) How to identify business applications of expert systems Second International Expert Systems Conference London October 1986

Suppes P (1979) Current trends in CAI Advances in Computers vol 18 Academic Press

Sussman G (1975) A computer model of skill acquisition Elsevier

Sviokla J (1986) Business implications of knowledge-based systems Data Base vol 18-1 Fall 86 p5-16

Szolovits P (1982) Artificial Intelligence in Medicine Westview Press Colorado

Tate A (1985) A review of knowledge based planning systems IN Merry (1985)

Taylor J and du Boulay B (1986) Why novices may find programming in Prolog hard Cognitive Studies Research Paper 60 University of Sussex

Taylor J and du Boulay B (1987) Learning and using Prolog Cognitive Studies Research Paper 90 University of Sussex

Tecuci G, Kodratoff Y, Brunet T and Bodnaru Z (1988) DISCIPLE: an expert and learning system IN Moralee (1988)

Thomas L F and Harri-Augstein E S (1985) Self organised learning Routledge and Kegan Paul

Thompson T F and Clancey W J (1986) A qualitative shell for process diagnosis IEEE Software March 1986 p6

Thorne M (1986) Call in the Expert MESU/CET

Todd R R (1988) Self organised learning within an intelligent teaching system IN Self (1988)

Toffler A (1970) Future shock Bodley Head

Toffler A (1980) The third wave Collins

Torrance S (1986) Ethics, mind and artifice IN Gill (1986)

Treadwell M (1986) ES/P Advisor in a Primary school IN Thorne (1986)

Turing A M (1950) Computing machinery and intelligence IN Feigenbaum and Feldman (1963)

Turkle S (1984) The second self: computers and human spirit Granada

Turner M (1985) A consultant's view of Expert Systems Data Processing vol 27-4 p12-14

Turner M (1986) Real time experts Systems International Jan 86 p55-57

Turner M (1988) Expert systems and decision support Expert Systems for Information Management vol 1-1 p3-21

Tulving E (1972) Episodic and semantic memory IN Tulving and Donaldson (1972)

Tulving E and Donaldson M (1972) Organisation of memory Academic Press

Van Melle W (1979) A domain-independent

production rule system for consultation programs Proc 6th IJCAI p923-925

Van Rijsbergen C J (1984) Research and development in information retrieval Cambridge University Press

Viccari R, Costa E and Coelho H (1987) A Prolog tutor for logic programming IN Nicol (1987)

Vickers G (1968) The art of judgement: a study in policy making Methuen

Vincent B and Vincent T (1985) IT and Further Education Kogan Page

Wallace M (1984) Communicating with databases in natural language Ellis Horwood

Walker A (1980) On retrieval from a small version of a large database Proc 6th International Conference on Very Large Databases Montreal Canada 1980 p47-54

Walker M G (1987) How feasible is automated discovery ? IEEE Expert Spring 1987 p69-82

Walton D (1986) Expert Ease in a Primary school IN Thorne (1986)

Wardropper J (1986) Computer Weekly 10/4/86 p17

Warnier J-D (1986) Computers and human intelligence Prentice Hall

Wason P C and Johnson-Laird P N (1968) Thinking and reasoning Penguin

Watson L (1987) The Prolog in Primary Schools Project IN Nicol (1987)

Watts S (1986) Surgeons held up money-saving report Computer Weekly 14/8/86 p3

Waterman D A (1986) A guide to Expert Systems Addison Wesley

Webb M (1987) Towards an expert system shell as a tool for conceptual and qualitative modelling in secondary education IN Nicol (1987)

Weiss S M, Kulikowski C A, Amarel S and Safir A (1978) A model-based method for computer aided medical decsion making AI vol 11 p145-172

Weiss S M and Kulikowski C A (1984) A practical guide to designing expert systems Rowman and Allanheld

Weizenbaum J (1966) ELIZA - a computer program for the study of natural communication between man and machine

Communications of the Association for Computing Machinery 9

Weizenbaum J (1976) Computer power and human reason: from judgement to calculation Freeman

Weizenbaum J (1984) Another view from MIT BYTE June 1984 p225

Welbank M (1983) A review of knowledge acquisition techniques for expert systems Martlesham Consultancy Services, Ipswich

Wenger T (1986) AI and tutoring systems MIT Press

Wheeler D K (1967) Curriculum process London Univ Press

Whittet M (1987) IT not guilty of job losses Computing 18/6/87 p4

Wiig K (1984) Market trends in AI in the USA and Japan IN Bernold and Albers (1985) p169-180

Wild M (1987) The computer and the use of Prolog in teacher education IN Nicol (1987)

Williams N (1986) The intelligent micro McGraw Hill

Winograd T (1972) Understanding Natural Language Edinburgh Univ Press

Winston P (1975) The psychology of computer vision McGraw Hill

Winston P (1977) Artificial intelligence Addison Wesley

Winston P and Horn B K (1981) LISP Addison Wesley

Winston P and Prendergast K (1984) The AI Business MIT Press

Wood S (1986) AI and theories of social situations IN Gill (1986)

Worden R (1988) Processes of knowledge and software Keynote lecture at Expert Systems '88

Wright W L, Green M W, Fiegl G and Cross P F (1986) An expert system for real time control IEEE Software March 1986 p16-24

Yazdani M (1984) New horizons in educational computing Ellis Horwood

Yazdani M (1986) AI and education: a critical overview IN Gill (1986)

Yazdani M (1987) AI tools for second language teaching IN Nicol (1987)

Yazdani M and Narayanan A (1984) AI: human effects Ellis Horwood

Young R M (1979) Production systems for modelling human cognition IN Michie (1979)

Zadeh L A (1979) A theory of approximate reasoning IN Hayes, Michie and Mikilich (1979)

Zarri G P (1984) Inference techniques in RESEDA IN van Rijsbergen (1984)

Zeide J S and Liebowitz J (1987) Using expert systems: the legal perspective IEEE Expert Spring 1987 p19-21

Zemankova-Leech M and Kandel A (1984) Fuzzy relational databases: a key to expert systems Verlag TUV Rheinland

Zigler E (1986) Intelligence: a developmental approach IN Sternberg and Detterman (1986)